KATHOLIEKE UNIVERSITEIT
LEUVEN

Arenberg Doctoral School of Science, Engineering & Technology
Faculty of Engineering
Department of Electrical Engineering (ESAT)

Privacy Preserving Content Protection

Mina DENG
邓米娜

Dissertation presented in partial
fulfillment of the requirements for
the degree of Doctor
in Engineering

June 2010

Privacy Preserving Content Protection

Mina DENG
邓米娜

Jury:
Prof. dr. ir. Hugo Hens, president
Prof. dr. ir. Bart Preneel, promoter
Prof. dr. ir. Joos Vandewalle
Prof. dr. Dave Clarke
Prof. dr. ir. Olivier Pereira
(Université catholique de Louvain)
Prof. dr. ing. Alessandro Piva
(Università di Firenze)

Dissertation presented in partial
fulfillment of the requirements for
the degree of Doctor
in Engineering

June 2010

Legal depot D/2010/7515/66
ISBN 978-94-6018-227-3

A journey of a thousand miles begins with a single step.
– Confucius (551 BC – 479 BC, China)

Preface

Over the past ten years, my life has undergone great changes. It took me from China to Belgium, and along the way, I have received a great amount of help and support from many people. I would like to take this opportunity to thank – at least some of – the people who have helped and encouraged me during the past years to realize this PhD dissertation.

First of all, I am grateful to my PhD promoter, Prof. Bart Preneel. Without him offering me the opportunity to join COSIC a few years ago, my doctorate wouldn't have been started in the first place. I appreciate the pleasant environment and the freedom that he created for me to do research in, the plenty of opportunities and funding that he offered for my trips to conferences and trainings, and his support and guidance through all my research activities.

I would like to express my gratitude to the members of my jury – Prof. Joos Vandewalle, Prof. Dave Clarke, Prof. Olivier Pereira, and Prof. Alessandro Piva – for taking time to review this dissertation and for their valuable feedback, and Prof. Hugo Hens for chairing the jury.

Prof. Vincent Rijmen, Prof. Ingrid Verbauwhede, and Dr. Claudia Diaz deserve my gratefulness for their advice and help to me on the numerous research and administration related issues. I would like to thank all my colleagues in COSIC as well, including the COSIC visitors I met, and especially my office mates for the good atmosphere. Our team has grown a lot over the past few years; sorry that I don't name all of you here. You have brought me a lot of pleasant and memorable experiences during my stay at COSIC. In particular, I would like to thank the members of the Privacy & IDM group and the SoPro group for the discussion sessions and joint learning activities in all these years, from which I have broadened my horizon and gained a lot of knowledge.

Throughout the course of my doctorate, I have had the pleasure to work together with several brilliant researchers. After years of fruitful collaborations, I have learnt a lot from them, and many of them have become friends. I'd like to acknowledge my co-authors: Danny De Cock, Alfredo Rial, Li Weng, Dr.

Riccardo Scandariato, Kim Wuyts, Prof. Wouter Joosen, Dr. Tiziano Bianchi, Prof. Alessandro Piva, Dr. Lothar Fritsch, Dr. Klaus Kursawe, Karel Wouters, Dr. Brecht Wyseur, and Thomas Herlea. Upon the completion of this dissertation, many friends have given me a helping hand in one way or another. I would like to thank especially, Nessim, Sebastiaan, Markulf, Elmar, and Stefan for their generous help with improving my text and preparing for the PhD defence.

I have spent a large amount of my time on a number of research projects. I would like to acknowledge all the colleagues, with whom I have collaborated, for their contributions that make the successful completion of these projects possible: Andreas, Antoon, Andras, Carmela, Christophe, Koen, Dave, Dennis, Dries, Sebastian, Stefaan, and Svetla.

Péla deserves my special thanks. Without her patience and readiness to help me, this dissertation would not have been realized. I also want to thank Elsy, Elvira, Annemarie, and Saartje for helping me out with the many administrative issues. You are indispensable in the COSIC family. Hartelijk bedankt!

I am indebted to my husband Stefan. Thank you for giving me unlimited power and the spirit of overcoming difficulties in the process of completing this dissertation. I want to thank my parents-in-law Joseph and Rina, and my brother-in-law Frederik. Thank you for creating a warm home for me. And last, but definitely not least, I would like to thank my parents. Thank you for inspiring my yearning for science and supporting my studies abroad from the beginning. Without your continuous support and encouragement, this doctorate would not have been accomplished.

Mina Deng
Leuven, June 2010

Abstract

The proliferation of information and communication technologies in the last few decades has profoundly influenced the way information is gathered and processed, and hence raised concerns over privacy. With the advent of digital technologies, the ease of unauthorized copying of and access to digital content leading to commercial infringements has motivated the need for developing content protection technologies. Content protection is a generalized term that essentially means restricting access to digital content. From a security perspective, it is to ensure confidentiality, integrity, and availability of content, in its distribution, reproduction and use.

The goal of this thesis is to investigate privacy issues in content protection systems and to explore techniques for privacy preserving content protection. The flourishing of content protection technologies enables a shift to an information environment characterized by pervasive constraints, universal monitoring, and automated processing. As a consequence, individuals' privacy rights are severely undermined. The fundamental attributes of content protection technologies determine the intrinsic conflict between preserving the interests of the content provider or copyright owner on the one hand, and protecting the privacy of the user on the other.

Our research addresses two issues. The first aims to lay out a generic and comprehensive framework to support the elicitation and fulfilment of privacy requirements. First, this framework provides a systematic methodology to model privacy-specific threats. An information flow oriented model of the system is leveraged to guide the analysis and to provide broad coverage. The methodology instructs the analyst on which privacy issues should be investigated, and where in the model those issues could emerge. This is achieved by (i) defining a list of privacy threat types and (ii) providing the mappings between the threat types and the elements in the system model. Second, this framework proposes an extensive catalogue of privacy-specific threat tree patterns that can be used to detail the threat analysis outlined above. Finally, this framework provides the means to map the existing privacy-enhancing technologies (PETs) to the identified privacy

threats, to facilitate the selection of sound privacy countermeasures.

The second part of our research aims to tackle the privacy protection issues in a number of specific content protection systems; these issues appear when limited trust is granted to the content or the service provider. Therefore, the proper balancing between content protection, for the content provider or service provider, and privacy protection, for the user, remains a research challenge. This research question is addressed from two aspects: (i) to design privacy preserving copyright protection techniques for protecting commercial content, for which we proposed anonymous buyer-seller watermarking protocols; and (ii) to design privacy preserving systems to manage and protect personal data using content protection techniques. These include a privacy-friendly architecture to manage distributed e-Health content, and a personal rights management system to enforce individual privacy rights.

Several conclusions can be drawn from this thesis. First, the research shows that just as user behaviour regulation has been chosen as one of the values of content protection technologies, so can privacy become one of the values embodied in content protection system design. In addition, the development of content protection technologies can respond to privacy protection requirements in a goal-oriented approach, such as following the proposed framework for privacy threat and requirement analysis, while complying with relevant legislation. Moreover, instead of impeding privacy, content protection technologies can be utilized to preserve and protect it. Finally, a number of insights for designers of privacy enhancing systems are provided; possible future research directions are discussed.

Samenvatting

De vooruitgang van informatie- en communicatietechnologie in de laatste decennia heeft de manier waarop informatie wordt verzameld en verwerkt grondig beïnvloed. Dit roept echter ook vraagtekens op rond de manier waarop met privacy wordt omgegaan. De komst van digitale technologieën, het gemak waarmee ongeoorloofde kopieën kunnen worden geproduceerd, en de toegang tot digitale inhoud – leidend tot commerciële inbreuken – motiveren de nood om *content protection* (inhoudsbeschermende) technologieën te ontwikkelen. Content protection is een algemene term die in feite het beperken van toegang tot digitale inhoud betekent. Vanuit een veiligheidsoogpunt dient het confidentialiteit, integriteit en beschikbaarheid van de inhoud te verzekeren in de distributie, reproductie en in het gebruik.

Het doel van deze thesis is om privacy aspecten van content protection systemen te onderzoeken en om privacy bewarende technieken voor content protection te verkennen. De opkomst van content protection technologieën maakt een verschuiving mogelijk naar een omgeving die gekenmerkt wordt door diepgaande beperkingen, universele monitoring, en geautomatiseerde verwerking. Als gevolg hiervan worden de privacy rechten van het individu zwaar ondermijnd. De fundamentele eigenschappen van content protection technologieën bepalen het intrinsieke conflict tussen het bewaren van de belangen van de content provider en eigenaar van het auteursrecht aan de ene kant, en het beschermen van de privacy van de gebruiker aan de andere kant.

Ons onderzoek belicht twee facetten. In het eerste deel wordt er een algemeen en uitgebreid kader voorgesteld ter ondersteuning van het bepalen en vervullen van de privacy vereisten. Eerst voorziet dit kader een systematische methodologie voor het modelleren van privacyspecifieke bedreigingen. Een informatiestroom-georiënteerd model van het systeem wordt aangewend om de analyse te begeleiden en te voorzien in een brede dekking. De methodologie schrijft de analist voor welke privacy kwesties zouden onderzocht moeten worden, en waar in het model deze kwesties zouden kunnen ontstaan. Dit wordt bereikt door (i) het definiëren van een lijst van privacy bedreigingen en (ii) het verstrekken van de

mapping tussen de types van bedreigingen en de elementen in het systeem model. Ten tweede stelt dit kader een uitgebreide catalogus voor van privacyspecifieke bedreigingsboompatronen die kan worden gebruikt voor het uitwerken van de hierboven geschetste bedreigingsanalyse. Ten slotte biedt dit kader de middelen om bestaande privacyverbeterende technologieën (*privacy-enhancing technologies, PETs)* te relateren aan de geïdentificeerde privacybedreigingen, om zodoende de selectie van solide privacy tegenmaatregelen te vergemakkelijken.

Het tweede deel van ons onderzoek heeft tot doel het aanpakken van de privacy beschermingsproblemen bij een aantal specifieke content protection systemen; deze kwesties verschijnen wanneer een beperkt vertrouwen wordt verleend aan de content- of de service provider. Het juiste evenwicht tussen content protection, voor de content of service provider, en bescherming van de privacy, voor de gebruiker, blijft een uitdaging voor verder onderzoek. Dit onderzoeksvraagstuk is gericht op twee aspecten: (i) het ontwerpen van privacybewarende technieken voor de bescherming van auteursrecht waarvoor we anonieme koper-verkoper watermerkprotocols voorstellen; en (ii) het ontwerpen van privacybewarende systemen voor het beheren en beschermen van persoonlijke data met behulp van content protection technieken. Deze omvatten een privacyvriendelijke architectuur voor het beheren van gedistribueerde e-Health data, en een persoonlijke-rechten beheersysteem voor het handhaven van persoonlijke privacy rechten.

Verschillende conclusies kunnen worden getrokken uit deze thesis. Ten eerste toont dit onderzoek aan dat net zoals het reguleren van het gebruikersgedrag gekozen is als een van de waarden van content protection technologieën, zo kan privacy een van de waarden worden die worden vooropgesteld bij het ontwerpen van content protection systemen. Daarenboven kan de ontwikkeling van content protection technologieën beantwoorden aan de privacybeschermingvereisten in een doelgerichte aanpak, zoals het volgen van het voorgestelde kader voor privacy bedreigings- en vereistenanalyse, terwijl de relevante wetgeving wordt nageleefd. Daarnaast kunnen content protection technologieën worden gebruikt om privacy te bewaren en te beschermen, in plaats van deze te verhinderen. Tot slot worden een aantal inzichten voor ontwerpers van privacy verbeterende systemen verstrekt; mogelijke toekomstige richtingen voor onderzoek worden besproken.

论文摘要

信息与通信科技在过去几十年的广泛发展，深刻地影响了信息收集与处理的方式，也加剧了大众对隐私保护问题的关注与担忧。方兴未艾的信息科技的发展，方便了信息的采集并促进信息的流动，实现了不用人为干预就能完成的信息过滤，进而降低了信息处理的成本。但是在这样做的同时，这种趋势也使用户的信息隐私权受到了威胁。

信息和通信技术的空前繁荣，为数字信息保护带来了严峻的挑战。数字信息可能有多种格式，包括文本文件与多媒体信息（比如图片，音频和视频）。随着数字技术的出现，新工具的不断完善，更便于以极低的成本制造出高质量的拷贝。未经授权的复制和访问数字信息导致商业侵权，促进了信息保护技术的研究和发展。

本论文旨在研究信息保护系统中的用户隐私权保护问题以及相关系统中的隐私增强技术。信息保护是一个广义的术语，本质上是指限制对数字信息的访问。从信息安全的角度来讲，它确保数字信息在其创作、发行、复制、储存和使用过程中的保密性、完整性和可用性。信息保护技术的保护机制可分为三类，即数字版权保护，复制保护，组播和单播媒体的有条件接入系统。数字信息保护技术的蓬勃发展逐渐形成了一个充满限制，广泛监控和自动化处理的信息环境。其结果是个人的隐私权利受到严重损害。信息保护技术经常被批评为侵犯个人隐私，因为它促使了针对个人知识习惯和喜好信息收集的大幅增加。由此可见，信息保护技术的基本属性决定了，一方面对信息提供者或著作权人利益的保护，与另一方面对用户的隐私保护之间利益冲突的存在。

本研究着重解决两个问题。第一个问题是制定出一个通用而全面的框架，以支持隐私保护要求的引出与实现。虽然信息隐私是我们社会所确定的优先事项之一，却很少有系统的有效的方法彻底分析隐私威胁。

首先，这个框架提供了一个系统的方法，为隐私的具体威胁创建模型。一个基于信息流的系统模型可以引导系统开发者的分析，并提供广泛的应用前景。该方法指出了系统开发者应该纠察的隐私侵权问题，提示在系统模型中的哪个环节可能出现此类隐私侵权问题。此性能可以通过两个步骤来实现：（一）定义隐私威胁的种类和目录，（二）提供隐私侵权威胁的类型与系统模型中的元素的映射。

其次，这个框架提出了隐私威胁树状模式的具体目录，以用于详细引导系统设计者进行上述的威胁分析。

最后，这个框架提供了这样一个方法：参照现有的隐私增强技术，来对应它们所解决的隐私威胁问题， 以实现隐私保护对策的有效选择。

研究的第二个问题旨在解决一些指定的信息保护系统中的隐私保护问题。这类问题根据的假设前提是当信息提供商或服务提供商被认定为有限的可信度时，新的隐私威胁就会出现。因此，如何找到为信息提供商或服务提供商提供数字信息保护服务和保护用户隐私间的适当平衡，仍然是一个挑战性的研究。对本问题的研究是从两个方面着手：（一）设计出针对商业信息的具有隐私保护性能的信息保护系统。为此我们提出了匿名的买方卖方数字水印协议，用于数字版权保护。（二）设计出针对个人数据的隐私保护系统，同时应用相关的信息保护技术。解决方案范例：包括一个管理分布式电子医疗信息并同时能维护病人隐私的电子医疗系统架构，以及一个执行个人隐私权的权利管理系统的框架。

本论文可以得出几点结论。首先，研究表明，正如用户行为监管已被列为信息保护技术的价值之一，隐私保护也可以成为信息保护系统的设计所体现出的价值。此外，隐私保护的设计要求可以通过系统的目标导向的方法，在信息保护系统中加以具体实现。例如，系统开发者可以利用本论文提出的框架，进行隐私威胁和需求分析，同时遵守有关法律法规。另外，信息保护技术可以被用来保护隐私，而不是妨碍它。本论文最后针对隐私保护系统的设计提出了几点见解，并对未来可能的研究方向进行了探讨。

Zusammenfassung

Informations- und Kommunikationstechnologie verbreitete sich in den letzten Jahrzehnten stark, dadurch wurde allgemein die Art und Weise der Informationsbeschaffung und -verarbeitung wesentlich beeinflusst. Dadurch werden Bedenken hinsichtlich des Datenschutzes hervorgerufen. Die inhärente Leichtigkeit, mit der auf digitale Daten auch ohne Berechtigung zugegriffen und mit der diese kopiert werden können, hat die Entwicklung von Inhalteschutzmechanismen nach sich gezogen. Unter Inhalteschutz verstehen wir dabei allgemein die gezielte Einschränkung des Zugriffs auf digitale Inhalte, aus Sicht der IT-Sicherheit muss dieser Vertraulichkeit, Integrität und Verfügbarkeit der Inhalte während deren Verbreitung, Vervielfältigung und Verwendung gewährleisten.

Ziel dieser Doktorarbeit ist es, Probleme von Inhalteschutzsystemen hinsichtlich des Datenschutzes zu untersuchen und Techniken zur Berücksichtigung des Datenschutzes im Inhalteschutz zu erforschen. Die universelle Verbreitung von Inhalteschutzmechanismen kann eine Veränderung der Informationsgesellschaft nach sich ziehen, welche durch tiefgreifende Einschränkungen bei der Informationsbeschaffung und -verarbeitung, allgemeine Überwachung und automatisierte Datenverarbeitung charakterisiert ist, wodurch das Recht auf informationelle Selbstbestimmung des Einzelnen ernsthaft untergraben würde. Die grundlegenden Eigenschaften von Inhalteschutzsystemen bedingen einen inhärenten Konflikt zwischen der Wahrung der Interessen des Inhalteanbieters oder Urhebers auf der einen und des Schutzes des Privatsphäre der Benutzer auf der anderen Seite.

Unsere Forschungsarbeit behandelt zwei Problembereiche. Im ersten zielen wir darauf ab, eine generische und umfassende Methodik zu entwickeln, die die Erfassung und Erfüllung von Datenschutzanforderungen unterstützt. Zum einen erlaubt diese Methodik die systematische Modellierung datenschutzrelevanter Bedrohungen, wobei ein informationsflussbasiertes Modell des Inhalteschutzsystems zur Steuerung der Analyse und zum Erzielen einer breiten Abdeckung genutzt wird. Dieses Vorgehen weist den Analysten darauf hin, welche Datenschutzprobleme wo im Modell zu untersuchen sind. Erreicht wird dies durch (i) die Klassifikation der Arten von Datenschutzbedrohungen und (ii)

das Verbinden dieser Bedrohungsarten mit den Elementen des Systemmodells. Zum anderen wird im Rahmen unserer Methodik ein umfassender Katalog von datenschutzspezifischen Bedrohungsbaummustern vorgeschlagen, welcher zur detaillierten Ausarbeitung der soeben beschriebenen Bedrohungsanalyse verwendet werden kann. Des Weiteren erlaubt es diese Methodik, den identifizierten Datenschutzbedrohungen bestehende privatsphäreunterstützende Technologien (englisch "Privacy-Enhancing Technologies", PETs) entgegen zu setzen, was die Auswahl geeigneter Gegenmaßnahmen erleichtert.

Der zweite Teil nimmt sich des Datenschutzproblems in einer Anzahl konkreter Inhalteschutzsysteme an. Unter der Annahme, dass neue Datenschutzbedrohungen immer dann entstehen, wenn den Inhalten oder dem Inhalteanbieter eingeschränktes Vertrauen entgegengebracht wird, stellt das Abwägen zwischen Inhalteschutz (aus Sicht des Inhalte- oder Dienstanbieters) und Datenschutz (aus Sicht des Benutzers) eine Herausforderung für die Forschung dar. Wir nähern uns diesem Forschungsthema aus zwei Perspektiven: (i) mit dem Ziel, den Datenschutz berücksichtigende Schutzsysteme für kommerzielle Inhalte zu entwickeln, wofür wir ein anonymes Käufer-Verkäufer-Wasserzeichenprotokoll vorgeschlagen haben; und (ii) durch den Entwurf privatsphärefreundlicher Systeme zur Verwaltung und zum Schutz persönlicher Daten auf Basis von Inhalteschutzsystemen, darunter eine privatesphärefreundliche Architektur für verteilte Inhalte in elektronischer Gesundheitssorge sowie ein persönliches Rechteverwaltungssystem zur Durchsetzung individueller Datenschutzrechte.

Aus dieser Doktorarbeit können einige Folgerungen gezogen werden. Zum einen demonstriert diese Forschungsarbeit, dass nicht nur die Reglementierung von Benutzerverhalten, welche häufig als zentrale Eigenschaft von Inhalteschutzsystemen gesehen wird, sondern ebenso auch der Datenschutz ein zentrales Anliegen solcher Systeme werden kann. Des Weiteren können mittels der vorgestellten Methodik zur Datenschutzanforderungs- und Datenschutzbedrohungsanalyse auf zielorientierte Weise gesetzeskonforme und privatsphärefreundliche Inhalteschutzsysteme entwickelt werden. Darüber hinaus wurde gezeigt, wie Inhalteschutzsysteme zum Schutz der Privatsphäre eingesetzt werden können, anstatt zu einer Bedrohung derselben zu werden. Schließlich werden für Entwickler zukünftiger privatsphärefreundlicher Systeme nützliche Hinweise erarbeitet und mögliche weiterführende Forschungsthemen besprochen.

List of Symbols and Acronyms

Sets and Elements

$c \in \mathcal{C}$	a ciphertext
$\mathcal{K}$	key space
$k \in \mathcal{K}$	key
$\mathcal{M}, \mathcal{C}$	plaintext space, ciphertext space
$m, p \in \mathcal{M}$	a message or a plaintext
$\mathcal{R}$	real numbers
$\{0,1\}^n$	bit-string of length n, i.e., any piece of digital data ($\{0,1\}^*$ indicates an arbitrary length).

Operators

$\odot_{\mathcal{M}}$	linear operator in the plaintext space
$\odot_{\mathcal{C}}$	linear operator in the ciphertext space
$\oplus$	the exclusive-or operation or addition modulo 2, or watermark embedding operation in the message space (depending on the context)
$\otimes$	denotes the watermark embedding operation in the encrypted domain
$+$	addition defined in the Galois field $GF(p^n)$
$\times$	multiplication defined in the Galois field $GF(p^n)$
$\oplus_c$	a function that adds c to the input (modulo 2) $\oplus_c(x) = x \oplus c$
$a \|\| b$	the concatenation of the strings/values a and b
$\|a\|$	bit-length of a bit-string a

Functions, Algorithms and Services

ψ	real world protocol
Anon	anonymization algorithm to issue a context-specific identifier from a global identifier
BDec	buyer's decryption, outputs a message m on input a ciphertext c and a secret key $sk_{\mathcal{B}}$
BEnc	buyer's encryption, outputs a ciphertext c on input a public key $pk_{\mathcal{B}}$ and a message m
BKeygen	buyer's key generation, outputs public key $pk_{\mathcal{B}}$ and a private key $sk_{\mathcal{B}}$
Check	algorithm to check whether the complaint from the seller is correct
Deanon	deanonymization algorithm to convert a context-specific identifier from a global identifier
Detect	algorithm to detect watermark from a pirated copy and to send the transaction record to a judge
D_k	decryption with key $k \in \mathcal{K}$
DocAnon	service to pseudonymize part of a document Doc
DocDeanon	service to convert a pseudonymized document back to the non-pseudonymized version
E_k	encryption with key $k \in \mathcal{K}$
$\mathcal{F}_{\psi}$	ideal functionality
$\mathcal{F}_{\mathrm{DRM}}$	an ideal functionality for a copyright protection protocol
$\mathcal{F}_{\mathrm{REG}}$	an ideal functionality for a registration protocol
GSgkg	group signature group key generation
GSiss	group membership issuing algorithm, interactive algorithms run by user $\mathcal{U}_i$ and issuer $\mathcal{I}$ respectively
GSjoin	group joining algorithm, for $\mathcal{U}_i$ to join a group
GSjudge	group signature opening verification algorithm
GSopen	group signature opening algorithm, outputs a pair (i, proof)
GSsig	group signature creation algorithm, outputs a signature s of a message m
GSukg	group signature user key generation
GSverify	group signature verification algorithm
IDConvert	service to convert context-specific identifiers
IDIssue	service to issue context-specific identifiers
Identify	algorithm to recover the buyer's identity
JDec	judge's decryption, outputs a message m on input a ciphertext c and a secret key $sk_{\mathcal{J}}$
JEnc	judge's encryption, outputs a ciphertext c on input a public key $pk_{\mathcal{J}}$ and a message m

JKeygen	judge's key generation, outputs public key $pk_{\mathcal{J}}$ and a private key $sk_{\mathcal{J}}$
Query	a query service
Request	algorithm run by a buyer to request the content from a seller
Response	algorithm run by a seller to deliver the content to a buyer
VerifyId	algorithm to verify the recovered identity
WATdet	watermark detection algorithm
WATemb	watermark embedding algorithm
WATsetup	watermarking setup algorithm
π_1	zero knowledge proof for fair encryption of private keys
π_2	zero knowledge proof for bit encryption

Parameters

$(\mathcal{B}_i, \tau)$	$\mathcal{B}$'s identity and its opening proof
$(i, proof)$	a proof *proof* that a member i has created the signature s
(pk, sk)	a public and privacy key pair
$(pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}')$	buyer's one time public and privacy key pair
$(pk_{\mathcal{J}}, sk_{\mathcal{J}})$	judge's public and privacy key pair
(s, m)	a group signature s of a message m
(upk, usk)	a user key pair
ϕ	a transaction index assigned by seller
$Aid_{\mathcal{X}_j}$	fixed-length context-specific identifier of entity $\mathcal{X}$ in a context j
b	an indicator of the correctness of the registration or deanonymization process
buyrequest	a buyer's request command (from $\mathcal{F}_{\text{DRM}}$ to $\mathcal{S}$)
C	encryption of the buyer's private key $sk_{\mathcal{B}}{}'$ to the judge
$cert_B$	$\mathcal{B}$'s certificate issued by $\mathcal{GM}$
ck	encryption of the secret watermarking key swk to the judge
contentID	content ID
crs	a request of the common reference string
ct	encryption of the final watermarked content Y
D	a distribution that parameterizes the ideal functionality of a registration protocol $\mathcal{F}_{\text{REG}}$
d	an indicator of a pirated copy or a verified signature (depending on contexts)
deanonym	a request of deanonymization
detect	a request of dispute resolution or watermark detection
detresp	a dispute resolution response command
$Did_{\mathcal{X}_j}$	patient $\mathcal{X}$'s electronic health record Doc_{A_j}'s document ID assigned by $\mathcal{H}_j$

$Doc_{\mathcal{X}j}$	patient $\mathcal{X}$'s electronic health record hosted by $\mathcal{H}_j$
$ew_{\mathcal{B}}$	$\mathcal{B}$'s encrypted watermark
$Gid_{\mathcal{X}}$	fixed-length global identifier of entity $\mathcal{X}$
gpk	group public key
gsk	group member's private signature key
$info$	a buyer's transaction record stored in a seller's record table
isk	private issuing key
j	an order of the item j from $\mathcal{B}$ that uniquely binds the transaction to X
K_{Dj}	the document ID provider DIP_j's secret key
K_{Docj}	the document anonymizer DA_j's secret key
K_e	symmetric secret key of the pseudo-random function
K_h	symmetric secret key of symmetric encryption function
K_{Pj}	the patient ID provider PIP_j's secret key
l_1	the bit-length of the index watermark ϕ
l_2	the bit-length of the watermark of buyer $W_{\mathcal{B}}$ or seller $W_{\mathcal{S}}$
$Loc(\mathcal{H}_j)$	location of $\mathcal{H}_j$
osk	private opening key
$pf_{sk_{\mathcal{B}}{}'}$	the proof of $\mathcal{B}$'s key escrow cipher C
$Pid_{\mathcal{X}j}$	patient $\mathcal{X}$'s context-specific ID in a context j assigned by $\mathcal{H}_j$
r	the common reference string
RAN	random number
Ref_j	variable-length context-specific reference of a context j
Reg	database held by the e-Health network registry Rg
reg_i	registration information stored in a registration table reg
register	a request of registration
regresp	a registration response command
release	a request to deliver the watermarked content
reqresp	a (buyer's) request response command
request	a request command (from $\mathcal{B}$ to $\mathcal{F}_{\mathrm{DRM}}$)
retrieve	a request of retrieving the registration information of a party P
Rights	content usage rights
σ	seller's random permutation function
sig_B	$\mathcal{B}$'s signature to pk_B signed with upk_i
swk	secret watermarking key
Sym	content symmetric key
$TabH_j$	database held by $\mathcal{H}_j$'s file repository FR_j
T_{tra}	seller's transaction record table
v	a registered value such as P's public key in $\mathcal{F}_{\mathrm{REG}}$
V	seller's index watermark
$\mathcal{W}$	a watermark space, with the watermark $W \in \mathcal{W}$
W	a watermark, or a composite watermark

$W_{\mathcal{B}}$	buyer's watermark
$W_{\mathcal{S}}$	seller's watermark
$W_{\mathcal{SB}}$	a composite watermark on inputs of buyer's and seller's watermarks
$\mathcal{X}$	an original content space, with the original content $X \in \mathcal{X}$
X	original content
X'	intermediate watermarked content
Y	watermarked content delivered to buyer, or a pirated copy of X

Parties

$\mathcal{A}$	real world adversary
$\mathcal{B}$	buyer
$\mathcal{C}$	challenger
CA	certification authority
CA-CoD	the compliance certificate issuer for the compliant device
CA-SC	the compliance certificate issuer the smart card
CP	content provider
$\mathcal{D}$	deanonymization authority
DA_j	document anonymizer of $\mathcal{H}_j$
DIP_j	document ID provider of $\mathcal{H}_j$
$\mathcal{E}$	simulator, ideal world adversary
FR_j	file repository of $\mathcal{H}_j$
$\mathcal{GM}$	group manager
$\mathcal{H}_j$	healthcare provider j
$\mathcal{I}$	issuer
$\mathrm{IDEAL}_{\mathcal{F}_\psi,\mathcal{E},\mathcal{Z}}$	ideal world ensemble
$\mathcal{J}$	judge
$\mathcal{O}$	opener
P	a generic entity/party
$\mathcal{P}$	a set of parties
PIP_j	patient ID provider of $\mathcal{H}_j$
$\mathcal{R}$	registration authority
$\mathrm{REAL}_{\psi,\mathcal{A},\mathcal{Z}}$	real world ensemble
$\mathcal{S}$	seller
$\mathcal{T}$	algorithm
$\mathcal{U}$	user
WCA	watermark certification authority
$\mathcal{X}$	patient, a medical document subject
$\mathcal{Y}$	registry in the e-Health network
$\mathcal{Z}$	environment

Abbreviations

AES	Advanced Encryption Standard
API	Application Programming Interface
BSW	Buyer-Seller Watermarking Protocol
CA	Conditional Access
CBC	Cipher-Block Chaining
CoD	Compliant Device
CP	Content Protection (or Copyright Protection)
DA	Document Anonymizer
DCRA	Decisional Composite Residuosity Assumption
DCT	Discrete Cosine Transform
DDH	Decisional Diffie-Hellman Assumption
DFD	Data Flow Diagram
DIP	Document Identity Provider
DPD	European Data Protection Directive 95/46/*EC*
DRM	Digital Rights Management, or Copyright Protection Protocol
EHR	Electronic Health Record
FR	file repository
GP	General Practitioners
GSM	Global System for Mobile Communications
HIPAA	Health Issued Insurance Portability and Accountability Act
HMAC	Hash-based Message Authentication Code
HVS	Human Vision Systems
ID	Identity, or National Identification Number
IDM	Identity and Information Management
IND-CCA2	Indistinguishability Under Adaptive Chosen Ciphertext Attack
IND-CPA	Indistinguishability Under Chosen Plaintext Attack
IOI	Item Of Interest
LINDDUN	Our Privacy Threat Taxonomy: Linkability, Identifiability, Non-repudiation, Detectability, Disclosure of Information, Content Unawareness, Policy and Consent Noncompliance
MUC	Misuse Case
NIST	National Institute of Standards and Technology
ODRL	Open Digital Rights Language
OMA	Open Mobile Alliance
OWASP	Open Web Application Security Project
p.p.t.	Probabilistic Polynomial Time
P3P	Platform for Privacy Preferences Project
PDA	Personal Digital Assistant
PET	Privacy Enhancing Technology

PII	Personal Identifiable Information
PIP	Patient Identity Provider
PK	Proof of Knowledge
PKI	Public Key Infrastructure
PRM	Personal Rights Management
QIM	Quantization Index Modulation
REG	Registration Protocol
SAML	Security Assertion Markup Language
SC	Smart Card
SCI	Smart Card Issuer
SDL	Security Development Lifecycle
SEI	Software Engineering Institute
STRIDE	Microsoft's Security Threat Taxonomy: Spoofing Identity, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege
TCPA	Trusted Computing Platform Alliance
TCP	Trusted Computing Platform
TCG	Trusted Computing Group
TTP	Trusted Third Party
XDS	Cross Enterprise Document Sharing
WGE	watermark generation and embedding phase
XrML	Extensible Rights Mark-up Language

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Privacy

The proliferation of information and communication technologies in the last few decades has raised the concern over privacy as an increasingly prevalent issue. The development of technology has brought major changes to the patterns of information dissemination and human interactions. At the same time, the ever-increasing capacity for information accessibility driven by technological innovations has brought massive opportunities in terms of abuse of privacy.

When arguing about privacy, scholars and theorists conventionally consider that privacy encapsulates three intertwined meanings, namely *physical privacy* (spatial seclusion and solitude), *informational privacy* (confidentiality, secrecy, data protection, and control over personal information), and *decisional privacy* (limited intrusion into decision making about sex, families, religion, and health care), as declared by privacy scholar Anita Allen [47]. Seclusion, solitude, secrecy, confidentiality, and anonymity are considered requirements for a liberal existence.

The development of these three kinds of privacy differs. The level of physical privacy – as a byproduct of increased wealth – in modern developed societies is extraordinarily high by historical standards. In addition, modern human rights and privacy legislation aids to grant and protect the civil liberties on the rights to make decisions, thereby ensuring decisional privacy prospers. However, the situation regarding informational privacy is worrisome. In the rest of the thesis, the term privacy and informational privacy are used interchangeably.

1.1.1 Technology and the Informational Privacy Concern

Emerging technologies ease the collection of information, facilitate bulk information flow, and enable filtering out information without human intervention, thus reducing the cost of processing it. We live in an era surrounded by information and computer technologies, ranging from surveillance and data mining, CCTV and image processing, Web 2.0 applications such as social networks, e-ID and e-government, e-commerce and e-health, to cloud computing, smart home, RFID, and location based services. They have profoundly influenced our way of living. However, many of them have caused a disastrous erosion of privacy, leading us to ask whether the age of privacy is over.

The low cost and the high capacity of information transmission and processing has resulted in a world where the all-encompassing information generated by cyber activities and online transactions is profiled and individualized. From an economic point of view, this stimulates the emergence and establishment of new markets in which the allocation and sharing of information is encouraged.

David Brin in his book *The Transparent Society* [81] depicts futuristic societies in which the burgeoning of surveillance technology seems unstoppable and privacy is overtaken. We will be limited to live with two choices: a world where those in power know everything about everyone, and a world where everyone is able to know everything about everyone including the ability to watch the watchers. Despite the similarities, these are disparate ways of life, representing the opposite relationships between citizens and their civic guardians. Unfortunately, neither case is fictional – the first is called *Surveillance* (literally meaning 'to watch from above'); the latter is referred as *Sousveillance* (stemming from the contrasting French words sur, meaning 'above', and sous, meaning 'below'). Sousveillance, coined by Steve Mann [177], is "a practice that originated with the use of eyetap (electric eyeglasses as described here) and other wearable computing devices, refers both to inverse surveillance, as well as to the recording of an activity from the perspective of a participant in the activity (i.e. personal experience capture)."

The privacy concern boils down to "what matters to me is not whether information about me exists but whether other people can find it. Modern information processing has at least the potential to drastically reduce that sort of (informational) privacy", said writer and scholar David D. Friedman [147], "if all information about you is readily available to anyone who wants it, you have no informational privacy. If nobody else knows anything about you, you have perfect informational privacy. All of us live between those two extremes." In view of privacy rights as control but not legal rules, Friedman stated that "privacy rights as commonly interpreted do not prevent people from giving out information about themselves, merely from obtaining information about others without their consent."

For example, the rise of social networking online implies that people no longer have sufficient informational privacy, according to Facebook's founder Mark Zuckerberg. Services of this world's most popular social network – such as Facebook's Beacon and its targeted advertising networks – have been considered a privacy nightmare [233, 160]. When talking about Facebook's privacy policies, Zuckerberg said privacy is no longer a "social norm". "People have really gotten comfortable not only sharing more information and different kinds, but more openly and with more people. That social norm is just something that has evolved over time." Arguing that online users have become more accustomed to sharing information online, such as on blogs and other social media services, Zuckerberg noted, "if I had created Facebook today, as opposed to several years ago, I would have made user information public, not private, by default as it was for years until the company changed dramatically in December (2009)" [247].

In response to the question whether Google's users should treat the search engine as a trusted friend, Google's CEO Eric Schmidt commented that "if you have something that you don't want anyone to know, maybe you shouldn't be doing it in the first place, but if you really need that kind of privacy, the reality is that search engines – including Google – do retain this information for some time...it's important, for example, that we are all subject in the United States to the Patriot Act and it is possible that all that information could be made available to the authorities" [297].

Concerning government surveillance and data mining, some believe that individual's privacy should yield to governmental or social demands. In the United States, for example, the 9/11 Commission Report to the Congress [257] recommended the "(1) presidential guidelines for information sharing, which 'should safeguard the privacy of individuals about whom information is shared'; and (2) guidelines confining the use of a government power which actually materially enhances security." The US principal deputy director of national intelligence Donald Kerr declared that "privacy no longer can mean anonymity. Instead, it should mean that government and businesses properly safeguard people's private communications and financial information" [176].

1.1.2 Debates over Privacy

The debate of the value of privacy concerning eavesdropping and surveillance originated in ancient times. Privacy has received an increasing attention, since the middle of the twentieth century. Scholar Deborah Nelson wrote in her book *Pursuing Privacy in Cold War America* [217], "since the end of 1950s the cry of 'the death of the privacy' has rung out from a wide variety of sources: journalism, television, film, literature, law enforcement, philosophy, medical discourse, and more". The pioneer privacy advocate Alan Westin in his 1967 book *Privacy and*

Freedom [289] already indicated that there is "a deep concern over the preservation of privacy under the new pressures from surveillance technology." Today, the problem of technology with the rising concern over privacy remains fundamentally the same.

Values of Privacy

Preserving the value of privacy against other social interests and norms has always been a controversial and contradictory proposition. Some assert that privacy nearly vanished; others argue that the threat to privacy is illusive. Some proclaim privacy to be inviolable; others advocate privacy can be eroded in needs of conflicting social values.

Legions of commentators and scholars have warned of diminishing privacy in modern societies. Privacy is recognized in a large part of the world as a fundamental right, essential for freedom, democracy, and human well-being. Professor and privacy feminist Anita Allen in her thought-provoking essay *Coercing Privacy* advances the propositions about *the liberal conception of privacy* (i.e., government ought to respect and protect interests in physical, informational, and proprietary privacy) and *the liberal conception of private choice* (i.e., government ought to promote interests in decisional privacy and allow individuals to make many of the important decisions concerning friendship, sex, marriage, reproduction, religion, and political association). According to Allen, "although the liberal conception of private choice is flourishing ... the liberal conception of privacy is not".

Privacy legal scholar Daniel Solove [274] pointed out the pluralistic nature of privacy's value, "privacy is a set of protections from a plurality of problems that all resemble each other, yet not in the same way. The value of privacy is not uniform, but varies depending upon the nature of the problems being protected against."

The opposite side argues that privacy is "detrimental, antisocial, and even pathological," and people are no longer interested in privacy since they "have nothing to hide" [273]. Legal scholar Fred Cate [92] describes privacy as an "antisocial construct" that "conflicts with other important values within the society, such as society's interest in facilitating free expression, preventing and punishing crime, protecting private property, and conducting government operations efficiently". Technological scholar Calvin C. Gotlieb [156] states that "most people, when other interests are at stake, do not care enough about privacy to value it", except for some outspoken "journalists, lawyers, and academics". Instead of privacy, Gotlieb suggests that people are interested primarily in confidentiality, regarding not the collection but the management of data. Judge Richard Posner [246] views privacy as the "right to conceal discreditable facts about himself". "When people today decry lack of privacy, what they want, I

think, is mainly something quite different from seclusion: they want more power to conceal information about themselves that others might use to their disadvantage". In other words, privacy is likely to be invoked when one needs to hide something that consists of negative information about a person.

The Privacy Tradeoff?

Appalled by today's global political environment, especially after the events of 9/11, many people in the US and some parts of the world have declared that they are willing to give up civil liberties such as privacy in the name of security, that this trade-off seems to be a "fait accompli" [265]. Due to its profound importance and increasing prevalence, both academia and popular media repeatedly debate the balance of security versus privacy: should it balance the degree to which an individual's privacy is compromised against potent national security interests? In such a debate, prevailing opinion was that a tradeoff exists between privacy and security: in order to get more of one, it will be at the expense of another. However, this fundamental dichotomy is questionable.

Many people tend to respond to the ever-increasing privacy destructive activities – such as government surveillance and data mining – with the argument that there is no problem because they have nothing to hide. "The nothing to hide argument is one of the primary arguments made when balancing privacy against security", analyzed Solove [273], "it is an argument that the privacy interest is generally minimal to trivial, thus making the balance against security concerns a foreordained victory for security". In other words, the argument reflects a comparison of the relative value of the privacy interest being threatened with the government interest in promoting security. Unfortunately, privacy is often traded for security or convenience. In reality, people routinely give out their personal information and willingly reveal intimate details about their lives online. In a 2009 poll in the US [277], 69% of the 551 surveyed users would consider having their identities verified when making an online purchase (e.g. by looking at incoming data such as the machine's unique hardware signature or its Internet address or less sophisticated authentication techniques such as cookies, or programs that collect browser data), on the condition that their personal data is not collected and the relationship with the vendors is trusted. About 75% of the surveyed users believe computer authentication is preferred for the convenience of remembering passwords or answering pre-selected personal questions.

When balancing privacy against security, instead of focusing the discussions on whether a particular information collection activity should be barred or not, both sides of the debate ought to consider the amount of oversight and accountability when the government engages in particular forms of information gathering. The fundamental problem of the "nothing to hide" argument and its variants is that it

conceives privacy in a singular and narrow way, focusing on one or two particular kinds of privacy problems (e.g. such as information collection and secondary use), while ignoring the other pluralistic aspects of privacy harm involved beyond exposing one's secrets to the government [273]. Solove [273] declared that "the security interest should not get weighed in its totality against the privacy interest." Alan Westin [289] believes that new technologies alter the balance between privacy and disclosure, and that the rights of privacy may limit government surveillance to protect democratic processes.

The US principal deputy director of national intelligence Donald Kerr [176] claimed that for "safety and privacy, I work from the assumption that you need to have both. These two components of security – safety and privacy – are the crux of much of what we're doing in the intelligence community." Security commenter Bruce Schneier [266] contests the privacy tradeoff with the statement that "security and privacy are not opposite ends of a seesaw; you don't have to accept less of one to get more of the other. The debate isn't security versus privacy. It's liberty versus control. If you set up the false dichotomy, of course people will choose security over privacy – especially if you scare them first. But it's still a false dichotomy. There is no security without privacy. And liberty requires both security and privacy."

1.1.3 Privacy Definitions

Privacy is a complex concept that encompasses profound and rich meanings. Without understanding the privacy definitions and problems, privacy cannot be addressed in a meaningful way. To meet the needs of understanding privacy in a comprehensive manner, there are a number of attempts to conceptualize privacy by a wide range of jurists, legal scholars, philosophers, psychologists, and sociologists. Privacy is an abstract and subjective concept, and the understanding and definition of privacy varies depending on social and cultural contexts, study disciplines, stakeholder interests, and application domains.

Privacy as Informational Self-determination

Dated back to the 1960s, Alan Westin's research is seen as the first significant work on the problem of consumer data privacy and data protection. Westin [289] defined privacy as: "*Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others.* Viewed in terms of the relation of the individual to social participation, privacy is the voluntary and temporary withdrawal of a person from the general society through physical or psychological means, either in a state of solitude or small-group intimacy or, when among larger groups, in a condition of anonymity or reserve." This definition is often referred as *informational self-*

determination in the literature. He further claimed privacy as a process that "each individual is continually engaged in a personal adjustment process in which he balances the desire for privacy with the desire for disclosure and communication of himself to others, in light of the environmental conditions and social norms set by the society in which he lives [289]."

According to Westin [289] privacy is described as four states: *Solitude*, meaning the individual is separated from the group and freed from the observation of other persons; *Intimacy*, meaning the individual is part of a small unit; *Anonymity*, meaning the individual while in public still seeks and finds freedom from identification and surveillance; and *Reserve*, meaning the creation of a psychological barrier against unwanted intrusion by holding back communications. Four basic functions of privacy in society are outlined by Westin [289], namely personal autonomy, emotional release, self-evaluation, and limited and protected communication.

Privacy as Individual Rights

Privacy rights are inherently intertwined with information technology. In response to new information and communication technologies, Warren and Brandeis declared that information which was previously hidden and private could now be "shouted from the rooftops", and privacy is referred as "*the right to be let alone*" [288], focusing on solitude and the freedom from intrusion, and protecting persons and their belongings from warrantless search.

Privacy as Access and Control

A variety of views on privacy have been proposed – such as characterizing privacy in terms of access and control – and in part summarized in [113]. Ruth Gavison [150] argued that interests in privacy are related to concerns over accessibility to others, and the concept of privacy is best understood as a concern for limited accessibility; one has perfect privacy when one is completely inaccessible to others. Privacy can be gained along three axes: secrecy (access to information about the person), anonymity (knowledge of the person's identity), and solitude (access to the physical proximity of the person).

Adam Moore [210], building on the views of Gavison and Allen, conceived privacy as the right of individuals to "*control access to oneself and to personal information about oneself*". He argues that privacy is relative to species and culture, and is objectively valuable. Moore claimed that privacy, like education, health, and maintaining social relationships, is an essential part of human well-being [113].

Privacy as Pluralistic Resemblances

Daniel Solove surveyed the criticisms of various scholars regarding each other's conceptions of privacy: although each of the privacy conceptions described above elaborates upon certain dimensions and contains countless insights, "settling upon any one of the conceptions results in either a reductive or an overly broad account of privacy" [271]. He pointed out that traditional theories attempt to define privacy by isolating a common denominator in all instances of privacy, that is, they seek to conceptualize privacy as a unitary concept with a uniform value that is unvarying across different situations. He argued that "the attempt to locate the essential or core characteristics of privacy has led to failure" [271]. Instead of conceptualizing privacy with the traditional method, Solove attempted to lay the groundwork for *a pluralistic understanding of privacy*, as "privacy is not reducible to a singular essence; it is a plurality of different things that do not share one element in common but that nevertheless bear a resemblance to each other" [274]. Privacy is articulated as a set of protections against a plurality of distinct but related problems. Each problem has elements in common with others but not necessarily the same – they share family resemblances with each other.

The methodology is from bottom up, as "a set of protections against a related cluster of problems", such that it will help resolve a wide array of privacy problems. Solove reconstructed privacy in four dimensions [274]: a *method* (to understand privacy as a plurality of things), a degree of *generality* (to work out contextually instead of in the abstract to provide a framework for understanding a broad range of privacy problems), a structure that accommodates *variability* (to leave room for significant variability in norms and cultures), and a *focus* (to focus on privacy problems). The framework to understand privacy in a pluralistic way will be reviewed as the privacy taxonomy in Section 1.1.4.

Privacy and Data Minimization

According to the European Data Protection Directive 95/46/EC [140], the principle of *data minimization* means that "a data controller should limit the collection of personal information to what is directly relevant and necessary to accomplish a specified purpose. They should also retain the data only for as long as is necessary to fulfill that purpose. In other words, data controllers should collect only the personal data they really need, and should keep it only for as long as they need it".

The data controller is "the person or administrative entity that determines the purposes and means of the processing of personal data on behalf of an institution or body" [140]. The data controller is responsible for the security measures protecting the data.

Data minimization requires that the possibility to collect personal data about others, the personal data collected, and the retention time of the collected data should be minimized. The main property of Privacy-Enhancing Technologies (PETs) is to limit the release of personal data and to preserve the unlinkability property as much as possible.

1.1.4 Existing Privacy Taxonomy and Notions

Taxonomy of Privacy

In 1960, legal scholar William Prosser synthesized the cases emerged from Warren and Brandeis's 1890 article *The Right to Privacy* [288], and discerned privacy as a legal concept to constitute four distinct torts [251]. These privacy torts are contoured as four categories: (1) intrusion upon seclusion or solitude, or into private affairs; (2) public disclosure of embarrassing private facts; (3) publicity which places a person in a false light in the public eye; and (4) appropriation of name or likeness.

In 2006, a taxonomy of privacy has been proposed by Solove [272, 274] as a framework for understanding privacy in a pluralistic and contextual manner from a social and legal perspective. Solove's taxonomy is an extension of Prosser's four categories of privacy torts, motivated by two reasons. One reason is that "Prosser focused only on tort law, and the law of information privacy is significantly more vast and complex." Another reason is that "Prosser wrote over 40 years ago, and new technologies have given rise to a panoply of new privacy harms."

According to Solove, the focus is shifted from conceptualizing the vague term *privacy* toward the specific activities that impinge upon privacy and pose privacy problems. With the ultimate purpose to aid the development of privacy law, this taxonomy is "an attempt to identify and understand the different kinds of socially recognized privacy violations" [272].

Solove's privacy taxonomy consists of four constitutive categories of privacy invasive activities: (1) information collection (surveillance and interrogation), (2) information processing (aggregation, identification, insecurity, secondary use, and exclusion), (3) information dissemination (breach of confidentiality, disclosure exposure, increased accessibility, blackmail, appropriation, and distortion), and (4) invasion (intrusion and decisional interference). Each group encompasses a variety of harmful activities as included between brackets. Each type of problem, and why it can be problematic, will not be discussed in this section due to space limits. We refer to [272, 274] for in depth explanations.

Ann Bartow [62] responded to Daniel Solove's taxonomy and criticized that it is "in failing to label and categorize the very real harms of privacy invasions

in an adequately compelling manner." In Bartow's view, the Solove taxonomy of privacy suffers from the lack of identifying "the compelling ways that privacy violations can negatively impact the lives of living, breathing human beings beyond simply provoking feelings of unease." Bartow pointed out that Solove "devotes substantially more energy to explaining causality than he does to explaining impact. This renders the taxonomy incomplete and unsatisfactory." Bartow suggested that "a more effective taxonomy would dramatically and thoroughly document the consequences of privacy violations in very visceral, dramatic ways."

Anonymity Terminology and Notions

A consolidated terminology of privacy has been proposed by Pfitzmann and Hansen [237]. It motivates and develops a number of definitions, including anonymity/identifiability, (un)linkability, (un)detectability, (un)observability, pseudonymity, partial identity and digital identity, and identity management. The new terminology differs from early definitions – such as ISO IS 15408 [3] – defining privacy in terms of data minimization. In addition, the relationships between these terms are described, and the defined properties are sketched.

This terminology is developed based on a setting where senders send messages to recipients using a communication network. It is therefore fair to state that the terminology is derived from perspectives related to anonymous communication. The terminology can be extended for other settings (such as users querying a database or customers shopping in an e-commerce shop) and derived by abstracting away the names of *sender*, *recipient*, and *message*. The definitions are made from the perspective of an attacker, who may be interested in monitoring which communication is occurring, which patterns of communication exist, or even in manipulating the communication. The attacker may be an outsider tapping communication lines or an insider able to participate in normal communications and controlling at least some entities. In addition, the privacy terminology is defined from a probability perspective. It is assumed that the attacker uses all information available to him to infer the probabilities of the items of interest (IOIs), such as who sent or received which messages.

To clarify the definition, a *subject* (or an entity) is regarded as a possibly acting entity such as a human being (i.e., a natural person), a legal person, or a computer. The setting that the terminology is based on can be concretely defined as a *system*, with a number of properties. First, the system has a surrounding, meaning that parts of the world are outside the system. In addition, the state of the system may change through actions within the system. We will follow the concept of subject and system in the rest of the thesis.

Hevia and Micciancio [161] formally defined a group of anonymity notions in the context of anonymous communication, in which users try to send messages to

Table 1.1. The anonymity notion by Hevia and Micciancio and the terminology by Pfitzmann and Hansen match between terms

Anonymity notion in [161]	Anonymity variant in [237]
	Relationship anonymity
	Relationship unobservability
Sender Unlinkability (SUL) (Σ, U)	Sender anonymity
Receiver Unlinkability (RUL) (U, Σ)	Recipient anonymity
Sender-Receiver Unlinkability (UL) (Σ, Σ)	Sender anonymity AND recipient anonymity
Sender Anonymity (SA) $(?, U)$	Sender unobservability
Receiver Anonymity (RA) $(U, ?)$	Recipient unobservability
Strong Sender Anonymity (SA*) $(?, \Sigma)$	Sender unobservability AND recipient anonymity
Strong Receiver Anonymity (RA*) $(\Sigma, ?)$	Recipient unobservability AND sender anonymity
Sender and Receiver Anonymity (SRA) $(\sharp, \sharp)$	
	Undetectability
Unobservability (UO) $(?, ?)$	Sender unobservability AND recipient unobservability

each other without revealing their identities. The notions capture the intuitive properties of anonymous channels defined in [237]. The framework proposed in [161] starts with the usual properties of communication networks, for instance, considering whether an attacker sees (U) the *values of the messages sent/received* for each sender/recipient, or only (Σ) the number of *messages sent/received* for each sender/recipient, or only $(\sharp)$ the *total number of messages*, or $(?)$ meaning *nothing*, as starting point to define several variants of anonymity. The anonymity notions proposed by Hevia and Micciancio [161] are compared with the terminology by Pfitzmann and Hansen [237] in Table 1.1. Take the associated mnemonic notation in the left column after the name and abbreviation: the first item of each pair describes what can be learned about each sender, and the second item describes what can be learned about each recipient.

The privacy notions that cover multiple anonymity and unlinkability variants were further extended by Bohli and Pashalidis [73], as Anonymity (AN), Strong

anonymity (SA), Weak anonymity (WA), Strong unlinkability with participation hiding (SUP), Strong unlinkability with usage hiding (SUU), Weak unlinkability (WU), Weak unlinkability with participation hiding (WUP), Weak unlinkability with usage hiding (WUU), and Pseudonymity (PS). The relations and structure of different notions is also completed, with the introduction of an application-agnostic hierarchy that presents potentially different degrees, where the correspondence between digital elements and users remains hidden from an adversary. Beyond those notions specific to anonymous communications as in [161], previously isolated privacy notions pertaining to group signature, anonymous communication, and secret voting systems are included in the hierarchy defined in [73], and thereby effectively made comparable. Bohli and Pashalidis also suggested [73] that it could be possible to place the privacy definitions pertaining to other system types, such as anonymous credentials, data anonymization systems, and sensor information systems into the privacy notions hierarchy. However, this remains a subject of future research.

1.2 Content Protection

In recent years, the unprecedented prosperity of digital and information technologies has introduced significant challenges to protect digital content in their creation, distribution, storage and use. Digital content may have a variety of formats, ranging from text documents to multimedia content, including image, audio and video.

Prior to the development of digital technologies, content was presented in analog forms, and the reproduction, usage, and distribution were at the price of content quality degradation. As a consequence, consecutive generational copies usually resulted in notable quality degradation, which in turn reduced the commercial value. This, together with existing analog copy protection techniques, discouraged illegal analog copying. In contrast, with the advent of digital technologies, new tools have emerged, enabling quality assured copies of the original content at a low cost. Digital media offers a number of advantages, including perfect reproduction, where copies produced are indistinguishable from the original content, and cost reduction for storage and distribution because of the efficient compression methods, where high-quality content can be stored on lower-capacity media and transmitted through lower-bandwidth networks.

Moreover, the development of the Internet has opened potentially limitless distribution channels for the electronic commerce of content. On one hand, this brings considerable advantages allowing more businesses to expand, costs to drop, and the introduction of personalized customer experience. On the other hand, the newly inspired opportunities raise the need for protecting the ownership rights of copyrighted digital content. The ease of unauthorized copying and

access to digital content motivates many commercial infringements. This was worsened with the Internet ever-increasing bandwidth and the capacity of digital storage, together with the worldwide digital communication and distribution mechanisms. Additionally, more illegal copies being reproduced and distributed automatically raises difficulties of tracing piracy across national borders and identifying individuals offering or receiving pirated content. These driving forces motivates the development of technological solutions for digital content protection, in compliance with corresponding legislation.

1.2.1 Basic Concept

Digital content protection is a generalized term that essentially means restricting access to digital content. From a security perspective, it is to ensure confidentiality, integration, and availability of content, in its distribution, reproduction and use. Content protection mechanisms conventionally target copyright protection, copy protection, and conditional access for multicast or unicast media. The major tasks that the content protection mechanism is expected to accomplish include, but are not limited to, copy protection, usage monitoring, distribution tracing, usage control, secure distribution of content and access keys, authentication of content source and receivers, association of digital rights with content, and renewability of content protection systems. These are the basic security requirements for an end-to-end content protection system, suggested by Arnold, Schmucker and Wolthusen [55], and by Eskicioglu and Delp [137].

Copy protection mechanisms guarantee that no additional replication is allowed than the permitted copies. Such schemes rely on the integrity of the copying devices and on specific device features to block illegitimate copying.

Usage monitoring mechanisms require that all usage information of content by users must be recorded or communicated to the content or rights owner. Such schemes mainly establish monitoring solutions for multimedia content, for instance, to determine royalty payments and the verification of broadcasting of commercials in accordance with contracts.

Distribution tracing mechanisms ensure that transmission information of content is created and recorded with features identifying the source or the destination of the transmission.

Usage control mechanisms require that the content or rights owner approves before any operation or use of the content. This implies the controlled access and usage facility.

Secure distribution of content and access keys means that multimedia content is compressed, packaged, and encrypted (e.g. with a symmetric cipher) in secure multimedia content distribution. Moreover, the decryption key for receiving

devices is usually encrypted (e.g. with a public key cipher) and distributed in a separate secure channel.

Authentication of content source and receivers refers to that before copyrighted content is transferred from one device to another, the source and receiving devices mutually authentication each other to provide evidence of licenses.

Association of digital rights with content implies that rights are embedded in the content using metadata or watermarks, or with right expression languages to express the rights of a party to a certain asset.

Renewability is needed to ensure forward security. The idea is to allow a content transmitter to identify the compromised devices and prevent the transmission of protected content, in case that legitimate devices are compromised to permit unauthorized use of content. For example, in the High-bandwidth Digital Content Protection (HDCP) system [131], each video receiver is issued a unique set of secret device keys (e.g. including a RSA private key), matched with a non-secret unique Receiver ID. The HDCP service provider (in this case, the Digital Content Protection LLC [38]) may determine that an HDCP Receiver's RSA private key has been compromised, and places the corresponding Receiver ID on a revocation list that the HDCP Transmitter checks during authentication. The HDCP Transmitter is required to manage system renewability messages (SRMs) carrying the Receiver ID revocation list. The validity of an SRM is established by verifying the integrity of its signature with the Digital Content Protection LLC public key. In this way, renewability allows an HDCP Transmitter to identify the compromised devices and prevent the transmission of HDCP Content, as specified in [131].

Another example of renewability is from the content protection system of – the next generation DVD – Blu-ray Disc. Blu-ray Disc's content protection system includes three primary components: *AACS*, advanced access content system, *BD+*, a Blu-ray specific enhancement for content protection renewability, and *ROM Mark*, a measure unique to Blu-ray Disc to guard against mass production piracy and sale of unauthorized copies of pre-recorded media. Blu-ray Disc employ the Advanced Access Content System, which provides different decryption keys, each of which can be invalidated should one of the keys be compromised. Revoked keys will not appear on future discs, rendering the compromised players useless for future titles unless they are updated to fix the issue. BD+ technology, developed by BD+ Technologies LLC [37], is based on the Self-Protecting Digital Content (SPDC) concept [141]. The goal is to prevent unauthorized copies of Blu-ray discs and the playback of Blu-ray media using unauthorized devices. BD+ is effectively a virtual machine in authorized players that can execute code included on discs to verify, authorize, revoke, and update players when needed. Since the content protection program is on the disc rather than the player, this allows for updating protection programs by simply having newer programs included on newer discs. Therefore, it gives content providers an additional means to respond to organized attacks on

the security system by allowing dynamic updates of compromised code [114].

With end-to-end security as an essential requirement, digital content needs to be protected in every stage of its lifecycle in order to prevent piracy losses and encourage the sustainable development of digital markets. One of the fundamental problems of an end-to-end content protection system is not only to ensure that the illegitimate consumers cannot access the requested content, but also to be capable of controlling how the content is used once it is in the user's possession.

1.2.2 Content Protection Core Techniques

Most content protection mechanisms rely on cryptography and multimedia security techniques. These techniques serve one or more of the requirements commonly sought after in information security and privacy from confidentiality through anonymity. In the following, an overview of the core techniques for content protection will be provided, and it is assumed that readers are familiar with basic principles of cryptography [204] and multimedia security technology [148].

Encryption and digital watermarking are recognized as the core techniques for content protection. *Encryption* [204] prevents unauthorized access to digital content, being the first line of defense and relying on reversible mathematical transformation based on a secret key. The limitation is that, once the content is decrypted, it does not prevent illegal replication by an authorized user.

Digital watermarking [102, 60], as the second line of defense, is a technique that allows some information to be embedded in digital content. Watermarking has a number of applications, as detailed in [60, 55]. When used for content protection for images or videos, the watermark embedded in the original content should be imperceptible and robust against various attacks.

As an application of watermarking, *fingerprinting* [60] can be used to identify the content and to associate it to a customer. The fingerprint can be either an intrinsic feature of the content or some external information embedded in it. At the algorithmic level, watermarking is the function that embeds this information, while fingerprinting refers to the complete protocol between content provider and a customer.

1.2.3 State of the Art of Content Protection Systems

In the past few years, a variety of content protection systems have been proposed and implemented in commonly used digital distribution networks. The five primary components to distribute digital content are satellite, cable, terrestrial, Internet, and prerecorded media including optical and magnetic media. Apart from this,

the components to store and process digital content in a digital home network (DHN) include a cluster of digital devices, such as PC, digital TV (DTV), digital video cassette recorder (DVCR), DVD player, set-top box and so on [139].

The state of the art content protection systems can be categorized into three groups, according to the taxonomy suggested by Eskicioglu and Delp [137], namely Digital Rights Management (DRM) systems – unicast- and multicast-based – for Internet distribution, Conditional Access (CA) systems for satellite, cable, and terrestrial distribution, and Copy Protection (CP) systems for distribution within digital home networks. Generally speaking, industry is particularly interested in developing digital content protection systems, including information technology, motion pictures, and consumer electronics. Note that this classification is not absolute. For instance, the content protection for IP/TV leads to merge of DRM and CA systems; the content protection mechanisms in Blu-ray Disc is in fact a hybrid of all three (DRM, CA, and CP) systems.

Regardless of the model used, the entities that involved in the content protection life cycle are the content owner, the content distributor, the customer with a compliant receiving device, and the clearing house. The *content owner* (sometimes as content provider) creates and packages – compresses, encrypts or watermarks – content according to established rules. The *content distributor* delivers content to consumers through distribution channels. The *customer* purchases and consumes content according to usage rules. The *clearing house* keeps track of financial and usage information. The following paragraphs provide an overview of three different models for content protection systems.

DRM Systems for Internet Distribution

Digital Rights Management (DRM) refers to the protection, distribution, modification and enforcement of the rights associated with the use of digital content [148]. Modern DRM Systems typically encrypt content with symmetric encryption; the content decryption key and usage rights are delivered in separate packages, separating the purchase and the delivery. Since the introduction in the mid 1990s, the research and development of DRM has been through ups and downs. It is a controversial topic from various viewpoints, such as the rules and controls imposed by the content industries contradicting the fair use and free speech principle for privacy. It is a multi-disciplinary subject where technology, law, and economics influence each other reciprocally; it is impossible to make scientific statements about DRM only from the view of one discipline without basing them on premises from the other disciplines. In order to focus the discussion, we mainly consider the distribution of digital content on the Internet, whereas the DRM systems within many companies' intranets are set aside.

DRM allows content providers to specify their own business model in managing the

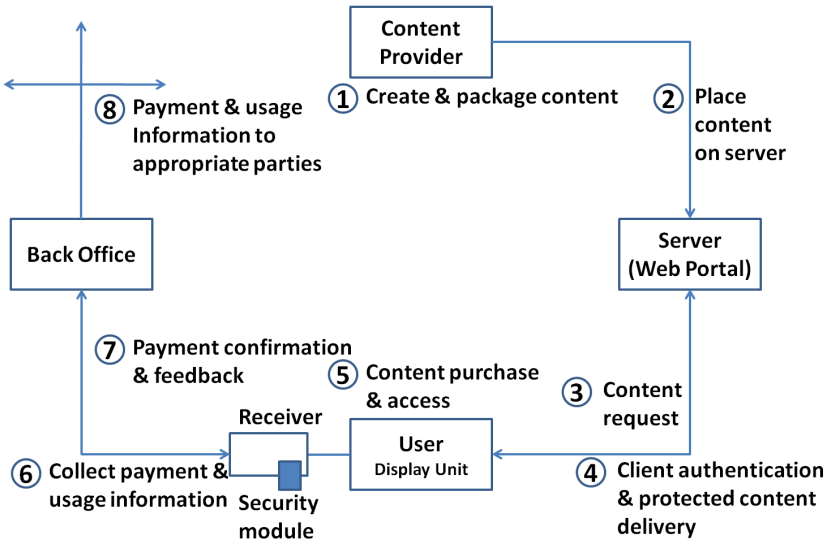


Figure 1.1. Architecture of Digital Rights Management systems for Internet distribution described in [195, 148]

use of the content, such as pay-per-use or time-limited use of content, e-commerce with superdistribution [211, 290], subscription, multiple views of a movie, and restrictions on transferring a song to a portable device, and so on.

From a technical point of view, a DRM system exploits cryptography – such as symmetric ciphers, public-key ciphers, and digital signatures – as the cornerstone for its security features, that include data confidentiality, key management and distribution, data source authentication, and security policies.

Despite the varying structure among DRM systems, a generic DRM system architecture for electronic delivery through the Internet [195] is shown in Figure 1.1. In this setting, it is assumed that there is a one-to-one communication (i.e. unicast) between the server and the customer's receiving device. The following steps summarize the information flow in a DRM-supported Internet distribution system:

1. The content provider (or seller) packages, such as inserting a watermark, and encrypts the content. The watermark may include information about the content provider, retailer, or extra information such as rights.

2. The protected content is placed on a server (e.g. connected with a web portal) for downloading or streaming. It can be located with a search engine using a proper content pointer.

3. The customer requests the content from the server, whereby a purchase transaction is usually required.

4. After the client device is authenticated, the content is delivered to the customer. Public key certificates are commonly used for the entity authentication process. Depending on the DRM system, the usage rules and the decryption key may either be pre-negotiated between the content provider and the customer or need to be separately obtained from the clearing house or other registration server in the form of a license. The content or other licenses are protected so that only the client is able to retrieve the information.

5. The customer obtains the content, decrypts it and uses it according to the negotiated rules.

6. The clearing house collects customer's payment records or usage history at certain times.

7. At the same time, payment and other information such as system and security updates are transmitted to the customer.

8. In the end, the customer's payment and usage information are sent to appropriate parties, including content providers and distributors.

This aforesaid unicast model can be extended to multicast networks in which content is delivered to a group of customers. In such a setting, three basic security features should be provided [148]. Primarily, only the legitimate group members have access to the current group communication. Additionally, legitimate group members should be able to authenticate the source and content of the group communication, in the case group members do not trust each other. Furthermore, it should support dynamic group management so that group membership can be granted or revoked whenever necessary. Similarly to the unicast case, the multicast systems also provide copyright protection for the content.

In both cases, compromised client devices or software will be included in the revocation list to allow servers to block content delivered to them. This, combining with updating DRM system components, ensures system renewability and forward security.

One of the most important issues of DRM technologies is interoperability. Most DRM systems work only in closed, monolithic systems that are designed not to be interoperable. It is monolithic in the sense that a DRM system typically supports a single protected content format and system for expressing and enforcing content usage rules. This non-interoperability can cause a series of negative effects, as declared by Ton Kalker [225]. Consumers are put off by content and services that do not work with all of their devices. Device manufacturers can choose

to either integrate a single DRM technology, thereby limiting the flexibility of their devices, or implement multiple DRM technologies adding to the cost of their devices. Content distributors are limited to choose DRM systems supported by popular devices, restricting their ability to address a broader set of consumers with different devices. Consequently, content providers see a lower business value due to the fragmented market.

In order to address the interoperability issue, a DRM interoperability framework *Coral* [14] was developed by a collection of content providers, service providers, consumer electronics manufacturers, and some technology companies. Fundamentally, the Coral Framework is designed to provide monolithic content distribution systems with a means to exchange any information that can enable the consumer to experience interoperability. The idea is to provide a standard DRM independent method for encoding a proof of purchase, e.g. using *Rights Tokens* [225], to minimize barriers to consumer access to content of all types, regardless of device, location, or time.

Conditional Access Systems

A generic architecture of Conditional Access (CA) systems [148] is shown in Figure 1.2. Such a CA system allows access to services based on payments or a number of security requirements, such as entity authentication (identification), data origin authentication, authorization, registration and so on. The content or service provider distributes content via satellite, cable or terrestrial transmissions to the end users with free-access to access-control such as Pay-TV and Video-on-Demand [138, 198, 134]. Instead of limiting ourselves to digital television systems which are often referred to, we broaden the definition to all sorts of application domains, such as services delivered through e-Health networks via cable [142, 71], as long as the functional setting fits in the general architecture.

This is the second approach for content protection; it is usually developed by the CA providers who are specialized in protecting both analog and digital content and in offering secure processing environments. The major components and common stages of a typical CA system are described as follows:

1. The content owner creates the content to be delivered to the CA provider, and compresses the digital content to minimize the communication bandwidth requirements.
2. The content is transferred to the CA provider in order to be protected and packaged with the entitlement information indicating access conditions.
3. The CA provider protects the content – for instance using encryption or watermarking – and delivers the protected content to the receiving device or a web portal connecting with a customer.

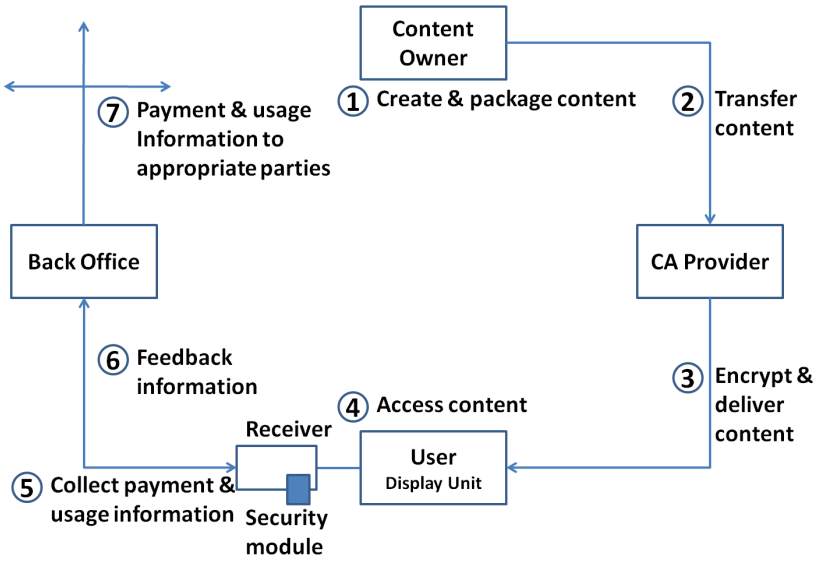


Figure 1.2. Architecture of Conditional Access (CA) systems described in [148]

4. If the customer is authorized to access the content, it is decrypted (or decoded) by the receiver (or decoder) and sent to the display unit for viewing. One of the critical tasks of a security module, such as a smart card, is to recover decryption keys. It also provides a secure environment to process security related information or functions such as authorization and temporarily stored purchase and usage information.

5. The billing and payment information, together with the collected purchase history and usage information are transmitted to a back-office, through a one-to-one link between the back office and the receiver.

6. In the meanwhile, information such as authorization and system security updates are delivered from the back office to the customer's receiver.

7. Finally, the collected payment and usage information are transferred to appropriate parties, such as the content owner, the CA provider, or other service operators.

Copy Protection Systems

Copy protection systems for distribution within digital home networks (composed of a cluster of digital devices) have the following aspects: (1) protection of content

on storage media, such as optical media, (2) protection of content across digital interfaces, such as Digital Transmission Content Protection (DTCP) and High-bandwidth Digital Content Protection (HDCP), and (3) management of rights associated with the content.

Unlike the aforementioned two approaches for content protection, the copy protection problem is considered to be the least promising to solve for a number of technical, legal and economic reasons [148, 153]. First of all, copy protection systems require devices and interfaces in home networks to be developed with a consensus among different stakeholders and manufactures. This makes the requirement determination and integration process difficult. Second, the ever-changing copyright legislation introduces controversial prohibitions subject to different interpretations. Finally, it is unclear who should pay for the copy protection in digital networks, and suitable business models are still under consideration.

Two groups of technologies are recognized as useful tools in designing solutions namely encryption-based and watermarking-based. As copy protection system structures vary among various application scenarios, instead of a general architecture, a collection of copy protection systems for optical and magnetic storage and two major digital interfaces [55] are briefly reviewed in the following paragraphs.

Copy protection solutions on optical media include Content Scramble System (CSS) for video on DVD-ROM [106, 61], where the CSS-protected video is decrypted during playback on the compliant DVD players; Content Protection for Recordable Media and Pre-Recorded Media (CPRM/CPPM) [2] for audio/video on DVD-R/RW/RAM, where A/V (audio/visual) content is re-encrypted before recording on a DVD recordable disk, and during playback the compliant player derives the decryption key; and a variety of watermarking schemes [199, 72, 9] for audio on DVD-ROM or video on DVD-ROM/R/RW/RAM, where invisible watermarks are embedded into the audio or video content, and the compliant playback or recording device detects the Copy Control Information (CCI) represented by the watermark and responds accordingly. Besides, the advanced access content system (AACS), BD+, and ROM Mark, are deployed copy protection mechanisms on Blu-ray discs [114].

Copy protection solutions on popular digital interfaces include Digital Transmission Content Protection (DTCP) [259] for the IEEE 1394 connection interface of A/V component communication and control, where the source device and the receiver authenticate each other and establish shared secrets; and High-bandwidth Digital Content Protection (HDCP) [131] for Digital Visual Interface (DVI) and High Definition Multimedia Interface (HDMI), where the video transmitter authenticates the receiver and they establish shared secrets. In both solutions, the A/V content is encrypted across the interface and the encryption key is renewed

periodically.

To enable copy protection and to define the conditions under which a consumer is authorized to make a copy, essentially, the Copy Control Information (CCI) is usually associated with the content. This can be achieved by either including the CCI in a designated field of the content, or embedding the CCI into the content as a watermark. For instance, the CCI makes use of the two Copy Generation Management System (CGMS) bits for digital copy control: "11" for copy-never, "10" for copy-once, "01" for no-more-copies, and "00" for copy-free. After an authorized copy, the compliant recorder could embed a new watermark to represent "no-more-copies". It is critical to assure the integrity of the CCI that unauthorized modifications can be prevented.

1.3 Privacy Issues in Content Protection

The proliferation of content protection technologies enables a shift to an information environment characterized by pervasive constraints, universal monitoring, and automated processing, which would severely undermine an individual's privacy rights. The fundamental attributes of content protection technologies create the intrinsic conflict between the basic starting point of preserving the interests of the content provider or copyright owner, and protecting the privacy rights of the user. The capabilities of content protection technologies have been criticized for implicating the user's privacy, by creating the potential for vastly increased collection of information about individual's intellectual habits and preferences [100, 235, 143].

The all-encompassing information aggregated and processed by content protection systems is typically personalized, such as one's transaction history, usage habits, purchasing behaviors or other profiling information. It places the customers a priori into an adversarial relation with the content provider. Moreover, the fact that information is at times gathered coercively or secretly, weakens the trust of customers in the content providers or service operators. It may further encourage customers to resist the content protection-engaged activities or deliberately provide false information. This in turn causes the content providers and service operators to pursue the information more aggressively, resulting in a vicious cycle.

Content protection systems collecting and monitoring an individual's information, on behalf of content providers or service providers, are often considered to be highly invasive for privacy of the individual, unless significant design and implementation efforts are expended to restrict such violations [143]. This is of particular concern in environments where content protection systems may be abused for surveillance, profiling, or similar privacy invasive activities, leading to the obstruction of

individuals' privacy rights. The limitations of portability and interpretability (e.g. rights are tied to specific devices that would require either a separate purchase or licensing to transfer the rights to another device), along with the possible loss of privacy due to the precise auditing and billing, can be viewed as placing the customer at a significant disadvantage, which may well balance or substantially outweigh any additional convenience for customers and, as a result, lead to an overall rejection of content protection support business models and the protected content [55].

1.3.1 Existing Privacy Enhancing Content Protection Systems

Early digital rights management systems for content delivery are device-based; they bind content to a device, and are considered overly restrictive, because it is difficult to transfer rights to other devices. Two approaches have been proposed to address this issue. The first approach is *domain-based DRM* [181] in which digital content can be transferred freely between the devices within the authorized domain. Deployed examples include Open Mobile Alliance DRM [7] and Apple's Fairplay [53]. The second solution is a user-centric approach named *person-based DRM* [181], where rights to access content are granted to users instead of devices to allow a user to access the content anytime, anywhere, and in any device. Deployed examples include Philips' Personal Entertainment Domain (PED) DRM [182]. In the following, two approaches to design a protection system are introduced: (1) privacy preserving content protection systems (for commercial content); (2) DRM-based content protection systems (for personal content) which is a person-based DRM system.

Privacy-Preserving DRM

A privacy-preserving DRM system (PPDRM) is described in [235]. These systems exploit (both persistent and short-term) user pseudonyms and provide a means of managing usage rights of commercial digital content, while preserving the privacy of users so that the content they purchased and the actions they took cannot be linked to a specific identity. The generalized architecture is shown in Figure 1.3.

Entities in the basic PPDRM system include the user, the content provider (CP) and the compliant device (CoD), a device that behaves according to the DRM rules. Related to the CoD, there is the compliance certificate issuer for compliant devices (CA-CoD). Moreover, there is the smart card (SC), which is the user ID device, and this entity is interchangeable with the user himself. Related to the smart card there are the smart card issuer (SCI) and the compliance certificate issuer for smart cards (CA-SC). Note that sufficient tamper resistance and detection mechanisms were provided in the original protocol as described in [235], but are omitted in the

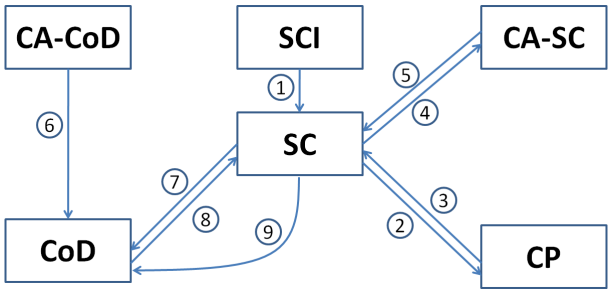


Figure 1.3. The basic architecture of the privacy-preserving DRM system described in [235]. The entity include the user and her smart card (SC), the content provider (CP) and the compliant device (CoD). Auxiliary entities are the compliance certificate issuer for the compliant device (CA-CoD), the smart card issuer (SCI) and the compliance certificate issuer the smart card (CA-SC)

following paragraph. For instance, each message described below is appended with its hash value and a signature, in order to ensure that the transferred messages have not been modified. The simplified transactions performed involving the basic PPDRM system are briefly described in the following steps:

1. The user acquires a smart card anonymously. The smart card is issued by SCI, to create a pair of user keys (pk, sk) with the user's secret PIN. Note that no one should be able to make an association between the user's real identity and the pk, and the private key sk is securely stored on the smart card and is not accessible to others (except for the SCI).

2. After an anonymous payment is made, the user's SC sends the public key pk to the CP.

3. The CP checks the legitimacy of the pk with the SCI and creates a license for that content. The content itself is encrypted by the CP with a symmetric key Sym, randomly chosen by the CP, and sent to the user.

4. The user's SC sends pk with a request for the compliance certificate to the CA-SC.

5. The CA-SC generates a pseudonym for the SC, say a random number RAN , and issues the compliance certificate to the SC.

6. The user's device CoD proves its validity by means of a CoD compliance certificate which is issued by the CA-CoD.

7. The license and the content are transferred from the SC to the CoD. Once the CoD has been checked, the SC proves its validity by showing the pseudonymous compliance certificate to the CoD.

8. The SC checks the compliance certificate of CoD, and receives the encryption $pk[Sym \parallel Rights \parallel contentID]$ of the content key Sym, the usage rights and the content from the SC.

9. The SC decrypts the message, and sends the values Sym, the rights and the content ID back to the CoD. The device CoD accesses the content.

In this PPDRM scheme, user privacy is achieved in the DRM system by decoupling the user's real identity from his identifiers (pseudonyms) in the DRM system (i.e., pk and RAN), or by decoupling the user's real identity from (contentID, Rights, Sym). Therefore, the real identity of the user cannot be revealed even by collusions of the parties above, since no party knows that identity; unless an attacker can obtain user-related information from the CoD after a content access transaction happens. This scheme essentially provides a similar privacy protection mechanism as that of the *direct anonymous attestation* of TPM (Trusted Platform Module) [80].

This PPDRM system can be extended to allow a user to further protect his privacy by purchasing content under different pseudonyms, such that unlinkability of user's transactions is assured. The various pseudonyms of the user must however be certified by a trusted authority (the CA) to guarantee security. This is achieved by pseudonym certification calculated from the user's public key pk by the CA.

DRM Technologies for Personal Content

The DRM concept can not only be used to protect commercial content, but also be exploited to design systems to protect personal content. In contrast to the significant effort put into the protection of copyrighted commercial content in DRM systems, controlled sharing of personal content is often ignored. To address this issue, a person-based DRM approach used for protecting ownership and controlled sharing of private content in home digital networks is presented [190]. The idea is to extend classical DRM systems for commercial content to protect private content. This effectively means that the user who is the content owner takes over the role of content and license provider, and therefore becomes involved in content creation and protection processes.

The privacy requirements in a content protection system have various aspects [190]: (1) users (as content owners) should be able to specify which content and to whom they will share; (2) the system must support co-ownership and manage the content sharing in various ways; (4) the system should support transfer of ownership; and (5) the users should be able to remove the content.

The proposed scheme uses a hybrid cryptographic approach, where a personal content is encrypted with a symmetric content key. The content key is encrypted

with the public keys of, and distributed to, the entities that have the access rights to the content. The above-mentioned privacy requirements are satisfied such that protecting and sharing personal content, ownership transfer, multiple ownership, and content or ownership deletion are supported.

1.4 Research Questions

This thesis aims to study privacy preserving content protection techniques and analyze private information flows in content protection systems. The research questions to be investigated in this thesis are divided into two parts, and are outlined in the following paragraphs.

1.4.1 Modeling Privacy Threats and Properties

The first part of the research focuses on providing a comprehensive framework to model privacy threats and requirements. There are a number of fundamental questions to be explored, including conceptualizing privacy properties and threats, evaluating the relation between privacy and security properties, and designing a generic methodology to identify privacy threats and elicit privacy requirements in application-dependent systems.

As an introduction, Section 1.1.4 provides an overview of a number of privacy taxonomies and notions developed in the literature. Solove's privacy taxonomy [272] identifies different kinds of socially recognized privacy violations and problems, from the social and legal viewpoint. The purpose of this taxonomy is to aid the development of the body of law that addresses privacy issues. It shifts the focus from the vague term privacy to the specific activities that pose privacy problems. The terminology by Pfitzmann and Hansen [237] provides definitions of privacy properties in review of anonymous communications; it is widely recognized and used in the community that researches privacy technical problems. The relation of privacy notions by Hevia and Micciancio [161] and its extension by Bohli and Pashalidis [73] essentially use the intuitive properties of anonymous channels defined in [237], and provide a formal proof of relations among privacy notions.

There is a gap in existing work, namely there does not include exist a taxonomic framework that captures privacy threats and identifies privacy properties to aid the development of a privacy enhancing system. The privacy properties and the derived threats should be taxonomized, based on the existing terminology and go beyond the scope of anonymous communication, to cover a broader concept of privacy, such as data or content privacy. This is motivated by a pragmatic reason – consider the fact that modern design of privacy enhancing systems is often accomplished using an ad-hoc approach. Regardless of the technical solutions

provided, during the design process, there is a need for a privacy taxonomy for system analysts and designers to identify the privacy property or evaluate privacy threats in the targeted system. As a systematic approach to build in privacy is needed, it is worth the effort to investigate an exhaustive and comprehensive methodology to map privacy properties to privacy threats, to identify the threats, and to elicit privacy requirements.

1.4.2 Designing Privacy-Friendly Content Protection Systems

The second part of the research focuses on exploring and designing individual privacy-friendly content protection systems. This question can be investigated from two viewpoints: (1) content protection systems with privacy preserving properties to protect commercial content, and (2) privacy preserving systems to manage and protect personal data using content protection systems or techniques.

The proper balancing between content protection and user's privacy protection remains a challenging issue. It raises a number of questions: how should the development and implementation of content protection technologies respond to privacy protection requirement and legislation? As user behavior regulation is already chosen to be one of the values of content protection technologies, could privacy also become one of the values embodied in content protection system design? How can content protection technologies be utilized to preserve and protect privacy?

Traditional content protection systems are based on the convention that content and service providers or rights owners are trustworthy, meaning that the adversarial model excludes such entities. Therefore, this kind of system focuses solely on protecting content and is hence limited to the provider-centric point of view. The customers (or users) have lost control of their private information.

A paradigm shift from provider-centric to user-centric takes place, that is, content providers are granted less trust and more incentives for privacy are desired by users. In a user-centric content protection system, content users are put back in the center of interest and are given control over personal information, for instance the users are able to specify which information (content) can be accessed by whom. This implies that users have the possibility to specify policies for how their personal information can be collected and processed.

To achieve this, it is necessary to enable privacy protection features in content protection technologies. For instance, anonymity or pseudonymity may constitute a requirement in content protection systems, i.e., the recipient of the content or the entity to which rights are granted may not be associable with an individuals' identity (either customer or content providers) unless it is associated with their pseudonyms.

The following paragraph briefly touches upon the controversial privacy issue inherent in content protection systems and highlights the two distinct strands of interest. In a DRM-supported Internet distribution system, there exists a contradiction between the content protection requirement – such as the content providers' desire for copyright protections and traceability for the copyright violator – versus individuals' privacy interests such as anonymity (or pseudonymity) and transaction unlinkability of customers.

In an e-health network, which can be viewed as a conditional access system, one of the contradictory issues is to provide an overview of a data subject's private information history (e.g. an overview of a patient's medication record) under access control versus restricted disclosure of the content subject's identity or sensitive information.

Another group of conflicting forces lays between the ease of broadly spreading personal content, such as images, and the rights of an individual to control the distribution of that very content. This problem rises against the background of widely used personal hand-held mobile devices which enable personal data to be conveniently collected and transmitted. Therefore, the adversary model is extended; and the adversaries not only include the classical ones, such as government and cooperate, but also ordinary citizens. To address this new privacy threat, there is the motivation to design and propose a personal rights management system.

This thesis will present selected privacy preserving solutions in an attempt to resolve the aforementioned contradicting forces between content protection (i.e., content providers trace and collect user information) and privacy protection (i.e., allow users to gain control over personal information), in order to enforce both the content provider's rights and the user's rights at the same time.

1.5 Outline and Summary of Contribution

This dissertation begins with an introduction to the privacy concerns in the information age, and motivates the necessity for content protection techniques, from which a number of privacy issues in content protection systems are pointed out. The rest of the dissertation will follow the research questions as outlined in Section 1.4. To facilitate the assessment of the research contribution demonstrated in this thesis, the following paragraphs summarize the contribution of each chapter. The content described in these chapters has been published in proceedings of peer-reviewed international conferences [125, 123, 129, 115, 116, 118, 128, 121, 122], international journals [130, 124, 258, 120], and book chapters [126, 119].

Chapter 2: Privacy Threat Analysis Framework. This chapter sets forth a

comprehensive and generic taxonomic framework to model privacy threats and elicit privacy requirements in application systems. Although informational privacy has become an identified priority in our society, few systematic and effective methodologies exist that deal with privacy threats thoroughly. First, this chapter provides a systematic methodology to model privacy-specific threats. Analogous to STRIDE in the Security Development Lifecycle to build in security, an information flow oriented model of the system is leveraged to guide the analysis and provide broad coverage for building in privacy. The methodology instructs the system analysts and designers on which privacy issues should be investigated, and where in the model those issues could emerge. This is achieved by (i) defining a list of privacy threat types and (ii) providing the mappings between threat types and the elements in the system model. Second, this chapter provides an extensive catalogue of privacy-specific threat tree patterns that can be used to detail the threat analysis outlined above. Finally, this chapter provides a guideline to map the existing privacy-enhancing technologies (PETs) to the identified privacy threats. Therefore, the selection of sound privacy countermeasures is structuralized.

Chapter 3: Anonymous Buyer-Seller Watermarking Protocols. This chapter investigates the privacy problems in a DRM system for Internet distribution between a seller (as the content provider) and a number of buyers (as the customers), and proposes a series of anonymous buyer-seller watermarking (BSW) protocols. Buyer-seller watermarking protocols integrate watermarking techniques with cryptography, for copyright protection, piracy tracing, and privacy protection. The proposed BSW protocol is based on homomorphic public-key cryptosystems and dynamic group signatures, and is able to provide all the required security properties, namely traceability, anonymity, unlinkability, dispute resolution, non-framing (i.e. malicious sellers cannot frame honest buyers), and non-repudiation (i.e. guilty buyers cannot deny having created illegal copies), simultaneously.

We have also worked out efficient implementations of BSW protocols, suggesting its practical relevance that this technique can be successfully used in practical applications. The implementation results are only briefly discussed in Appendix C. More details can be found in [129, 115, 116, 117].

Chapter 4: Privacy Friendly Architecture to Manage Distributed Personal E-Health Information. This chapter lays out an architecture to manage distributed personal e-Health information, in order to address the contradiction between privacy protection for the patient and the distribution of medical data. Primarily, the goal of such an e-Health system is to provide a patient-centric lifelong view on medical history under conditional access. This requests the authorized accessibility to a particular patient's healthcare information anytime, anywhere, on any device. Considering the distinguishing features of e-Health systems, especially, that health data are sensitive by nature, a privacy protection mechanism is proposed which limits the trust in service providers, in order to

facilitate a privacy enhancing sharing of distributed e-Health information.

Chapter 5: Personal Rights Management for Individual Privacy Enforcement. This chapter introduces the concept of Personal Rights Management – an architecture that manages personal data and enforces individual privacy rights. With the ubiquitous use of digital camera devices, especially in mobile phones, privacy is no longer threatened by governments and companies only. The new technology creates a new threat by ordinary people, who could take and distribute pictures of an individual with no risk and little cost in any situation in public or private spaces. Fast distribution via web-based photo albums, online communities and web pages expose an individual's private life to the public. Social and legal measures are increasingly taken to deal with this problem, but they are hard to enforce in practice. This chapter proposes a model for privacy infrastructures intended towards the distribution channel such that as soon as the personal content is publicly available, the exposed individual gets an opportunity to find it and take appropriate action without any delay. Digital rights management techniques were applied in the proposed infrastructure, and the use of data identification techniques such as digital watermarking and robust perceptual hashing were proposed to improve the distributed content identification.

Chapter 6: Conclusion and Future Research. This chapter draws conclusions from the aforementioned research questions, and discusses future research directions to further strengthen and deepen the research presented in the previous chapters.

Collaboration

The research work presented in Chapters 2, 3, 4 and 5 has been carried out in collaboration with fellow researchers Danny De Cock, Alfredo Rial, Li Weng and Prof. Bart Preneel from COSIC, Dr. Riccardo Scandariato, Kim Wuyts and Prof. Wouter Joosen from the DistriNet group of K.U.Leuven, Dr. Tiziano Bianchi and Prof. Alessandro Piva from University of Florence, Dr. Klaus Kursawe, then a research fellow from COSIC and now a research scientist at Philips Research Labs in Eindhoven, and Dr. Lothar Fritsch, then a research fellow from Johann Wolfgang Goethe University in Frankfurt and now a research scientist at Norsk Regnesentral (Norwegian Computing Center) in Oslo. Many research ideas and improvements are inspired through interactive discussions with them, therefore they deserve the credits. Moreover, I have provided fundamental contributions and a substantial amount of work in completion of all the aforementioned research.

The work described in Chapter 2 and 4 was respectively funded by the E-HIP (E-Health Information Platform) [24] and the Share4Health (Healthcare professional's collaboration Space) [39] research project from IBBT (Interdisciplinary Institute for Broadband Technology). The results elaborated in Chapter 3 and 5 were mainly

carried out and supported by the SPEED (Signal Processing in the EncryptEd Domain) research project [35] and the FIDIS (Future of IDentity in the Information Society) Network-of-Excellence [32] funded by European Commission under the FP6, respectively.

Chapter 2

Privacy Threat Analysis Framework

2.1 Introduction

Privacy becomes increasingly important in the current society. Most of the information is now digitalized to facilitate quick and easy access. It thus becomes important that this digital privacy is sufficiently protected to prevent personal information from being revealed to unauthorized subjects. A stepping stone of security and privacy analysis is threat modeling, i.e., the "black hat" activity of looking into what can possibly go wrong in a system. Threats are crucial to the definition of the requirements and play a key role in the selection of the countermeasures. Methodologies and knowledge are two important pillars for security and privacy requirements engineering [202]. Unfortunately, the state of the art lacks systematic approaches to model privacy threats, elicit privacy requirements, and instantiate privacy-enhancing countermeasures, accordingly. Indeed, there is an asymmetry for privacy with respect to security concerns. These latter have a far better support in terms of methodological approaches to threat modeling. For instance, in the goal-oriented requirements space, KAOS provides a methodology to systematically analyze a system's anti-goals (and the corresponding refined threats) and therefore derive security requirements [283]. The same holds in the area of scenario-based techniques. For instance, Microsoft's STRIDE is an industrial-level methodology to eliciting threat scenarios and, thence, deriving security use cases [163]. Notably, a significantly sized body of reusable knowledge is also available in the secure software engineering community. Security knowledge is often packaged in the shape of checklists and patterns. For instance, STRIDE comes bundled with a catalogue of security threat tree patterns

that can be readily instantiated in the system at hand so as to elicit a close-to-exhaustive set of potential security threats. Unfortunately, privacy is still lagging behind.

2.1.1 Previous Work

Privacy Requirements Analysis

Mylopoulos et al. [213] are the first to point out that complexity of an information system is determined partly by its functionality (i.e. what the system does) and partly by its non-functional requirements (also referred as constrains, goals, and quality attributes), and the non-functional requirements "play a crucial role during system development, serving as selection criteria for choosing among myriads of decisions". They proposed a comprehensive framework for representing and using non-functional requirements during the development process in a process-oriented approach. The framework consists of five basic components – goals, link types, methods (i.e., goal decomposition methods, goal satisficing methods, and argumentation methods), correlation rules, and the labeling procedure – as the representation of non-functional requirements in terms of interrelated goals. As suggested by Mylopoulos et al. "such goals can be refined through refinement methods and can be evaluated in order to determine the degree to which a set of non-functional requirements is supported by a particular design" [213]. This framework can serve as the foundation for the proposed privacy requirement analysis methodology.

The privacy guidelines provided by Microsoft describes some basic privacy concepts [29], such as different types of consents or data minimization concepts. Besides, a number of guidelines are presented for selected scenarios concerning the following principles: notice, choice, onward transfer, access, security, and data integrity. However, it only contains a flat list of the required and recommended guidelines and does not intend to describe a more structured approach. These guidelines can still be used as inspiration to determine possible threats, for example, to extend our catalogue of threat trees.

In the documentation of SDL version 3.2 [30], privacy is also partially considered, but only at a generic level. For example, the threat modeling process described in the forth stage of SDL only mentions that a design review with the privacy expert is necessary. SDL also presents ten general privacy guidelines. An example guideline indicates that it is important to collect the least sensitive form of data. At this stage, privacy is not yet well integrated in SDL.

Yu and Cysneiros [298] presented a framework using $i*$, an agent-oriented requirements modeling language, to deal with privacy requirements. This framework however focuses on reasoning about privacy and does not provide

a structured methodology to examine the different privacy objectives. Liu et al. [194] proposed a framework also using $i*$ to deal with security and privacy requirements, which was inspired by the work of Yu and Cysneiros. They use four different analysis techniques to create a complete model: attacker analysis, dependency vulnerability analysis, countermeasure analysis and access control analysis. Although this framework contains the attacker analysis technique which is similar to our idea of examining the possible privacy threats, the framework is again mainly meant to reason about privacy but lacks the necessary knowledge to empower the (non-expert) privacy analyst.

Miyazaki et al. [207] defined a computer-aided privacy requirements elicitation technique. This technique returns the appropriate requirements for the system to be compliant with the law, based on a questionnaire that the system engineer fills out. This is in contrast to our methodology, which helps the analyst eliciting the privacy requirements in accordance to the stakeholders' wishes.

From Privacy Requirements to Privacy Solutions

Several taxonomies have been proposed to create a link between privacy requirements and privacy solutions. This section gives an overview of some existing taxonomies which can be integrated in our threat modeling process to link the privacy requirements obtained by our methodology to the optimal privacy enhancing solution(s).

The taxonomy of privacy goals, described by Antón et al. [52], is to analyze website privacy requirements. Privacy goals are divided into protection goals and (anti-) vulnerability goals. The Code for Fair Information Practices is used to categorize the protection goals, namely notice and awareness, choice and consent, access and participation, integrity and security, and enforcement and redress. The (anti-) vulnerability goals are classified according to the manner in which they violate the user's privacy. The corresponding goals are monitoring, aggregation, storage, and information transfer. These goals are used to analyze and compare privacy policies and do not intend to link privacy requirements to general privacy solutions, instead, limited to website privacy requirements.

The Pris method [173] presents a structured way to create systems which adhere to the specified privacy requirements. It consists of four different phases: 1) Elicit privacy-related goals, 2) Analyze the impact of privacy goals on organizational processes, 3) Model affected processes using privacy-process patterns, and 4) Identify the technique(s) that best support or implement the above processes. The last step uses a table that classifies privacy implementation techniques in six categories, a) Administrative tools, b) Information tools, c) Anonymizer products, services and architectures, d) Pseudonymizer tools, e) Track and evidence erasers,

and f) Encryption tools, and it can be an interesting source of inspiration to determine appropriate privacy solutions.

Wuyts et al. [294] created a hierarchical taxonomy which categorizes privacy into objectives. The objectives are divided in two branches. The proactive branch focuses on concealing the association between the data and the identity of the user before it is shared with the system, while the reactive branch focusses on guarding the relationship between the data and the identity when the data is already shared. Each objective corresponds to a number of strategies, which are a sub-classification of the objectives. These strategies can then be linked to their corresponding solutions. This paper only provides a sample solution for each category and does not provide a full overview of all existing solutions.

2.1.2 Summary of Contributions

This chapter contributes to the aforementioned dimensions, both from methodologies and knowledge review points, by providing a comprehensive privacy threat modeling framework to support the elicitation and fulfillment of privacy requirements.

First, this chapter provides a systematic methodology to model privacy-specific threats. Analogous to STRIDE, an information flow oriented model of the system is leveraged to guide the analysis and to provide broad coverage. The data flow diagram (DFD) notation has been selected, as described in Section 2.2. The methodology instructs the system analyst or designer on what issues should be investigated, and where in the model those issues could emerge. This is achieved by defining a list of privacy threat types and providing the mapping of the threat types and the elements in the system model. This part of the methodology is described in Section 2.5. Note that the privacy threat types have been identified in contrast with well known privacy objectives, as summarized in Section 2.4.

Additionally, this chapter provides an extensive catalogue of privacy-specific threat tree patterns that can be used to detail the threat analysis outlined above. In a nutshell, they refine the privacy threat types by providing concrete examples. The catalogue is described in Section 2.6, while Section 2.7 illustrates how to instantiate the threat tree patterns in order to elicit the misuse cases.

Finally, this chapter provides the means to map the existing privacy-enhancing technologies (PETs) to the identified privacy objectives. Therefore, the selection of sound privacy countermeasures is simplified. This is described in Section 2.8.

The work described in this chapter has been published in [130].

2.1.3 Chapter Outline

This chapter is organized as follows. Section 2.2 briefly reviews the preliminaries including the Data Flow Diagram to model application systems and the security threat modeling framework STRIDE. An overview of our proposed LINDDUN methodology for modeling privacy threats, eliciting privacy requirements, and selecting privacy enhancing technologies accordingly is provided in Section 2.3. Section 2.4 summarizes the privacy objectives, from which the privacy threat types can be identified. Section 2.5 further elaborates the LINDDUN methodology by introducing the privacy threat categories based on the privacy properties and the mapping of these categories to the DFD elements. An extensive catalogue of privacy-specific threat tree patterns are provided in Section 2.6, whereas the instantiation of the threat tree patterns to elicit the misuse cases is described in Section 2.7. The elicitation of privacy requirements from the documented misuse cases is explained in Section 2.8.1. Section 2.8.2 proposes a number of privacy enhancing countermeasures to the identified privacy threats in a software or hardware based system. The proposed methodology is discussed in Section 2.9, and the chapter concludes in Section 2.10.

2.2 Preliminaries

2.2.1 Data Flow Diagram

An application system can be graphically represented using Data Flow Diagrams (DFD) that consists of the following elements: data flows (communication data), data stores (logical data or concrete databases, files, and so on), processes (units of functionality or programs) and external entities (end-points of the system like users, external services, and so on). For threat modeling, trust boundaries are also introduced to represent the border between trustworthy and untrustworthy elements.

Let's take a Social Network 2.0 use case as a running example to be discussed throughout this chapter. The DFD is an abstract representation of a social network application, as depicted in Figure 2.1, where online users share personal information such as relationship status, pictures, and comments with their friends. In this DFD, the user is represented as an entity to interact with the system. The Social Network 2.0 application contains two processes (the portal and the service) and one data store which contains all the personal information of the users. The trust boundary shows that the processes, the data store, and the communication between the two are considered trustworthy.

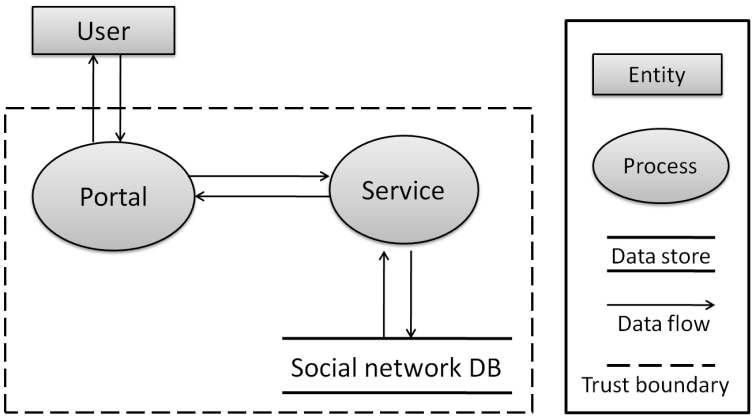


Figure 2.1. The Data Flow Diagram (DFD) of a Social Network 2.0 application

2.2.2 Security Threat Modeling – the STRIDE Approach

Security, in contrast to privacy, has already been well integrated in the Security Development Lifecycle (SDL) [163] as a well-established methodology. The SDL process consists of a set of activities that can be integrated in each step of the development lifecycle of application systems. The main advantage of the SDL approach is the reduction of the number of security vulnerabilities and the total cost of the system development, by eliminating security vulnerabilities at an early stage of development.

To build security in software or hardware based systems, an important aspect is to consider how an attacker might compromise the system security by exploiting design flaws and building the necessary countermeasure mechanisms in the system. In this respect, the threat modeling plays a key role. In fact SDL has integrated a systematic approach for security threat modeling called STRIDE. In this section, we will briefly review the STRIDE threat modeling process, and its nine key steps are summarized in the following paragraphs.

1. *Define use scenarios.* As a first step, system's key functionalities will be determined and use case scenarios will be defined.

2. *Gather a list of external dependencies.* A number of external dependencies need to be defined, such as the operation system that the system functionality depends on, the database it uses, and so on.

3. *Define security assumptions.* In the analysis phase, often decisions are based on implicit assumptions, which may not always apply anymore after

Table 2.1. Security concerns with corresponding security threats and DFD elements susceptible to threats (DF-Data flow, DS-Data store, P-Process, E-External entity), proposed by the Security Development Lifecycle (SDL)

Security property	Security threat	DF	DS	P	E
Authentication	Spoofing			×	×
Integrity	Tampering	×	×	×	
Non-repudiation	Repudiation		×	×	×
Confidentiality	Information Disclosure	×	×	×	
Availability	Denial of Service	×	×	×	
Authorization	Elevation of Privilege			×	

a second iteration, therefore, it is important to explicitly note down all the assumptions, to have a good understanding of the entire system.

4. *Create external security notes.* Because each external dependency can have its view on security, it is useful to list all the restrictions/decisions made by the external dependencies. An example of such a security note is which ports are open for database access or HTTP traffic.

5. *Create one or more DFDs of the application being analyzed.* The system is decomposed in relevant (either logical or structural) components and for each of these parts the corresponding threats are analyzed. This process is repeated over increasingly refined model until a state is reached where the residual threats are acceptable.

6. *Determine threat types.* The STRIDE threat taxonomy is used to identify security threat types. STRIDE is an acronym for Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege. These threats are the negation of the main security properties, namely confidentiality, integrity, availability, authentication, authorization and non-repudiation.

7. *Identify the threats to the system.* Each element of the data flow diagram is assigned to a set of susceptible threats. Table 2.1 gives an overview of the different DFD elements with the corresponding security threats they are subject to (marked with ×).

   To identify which threats are applicable to a specific system, threat tree patterns can be used. For each valid intersection in Table 2.1, a threat tree pattern suggests the possible security-related preconditions for the STRIDE category, in order to help analysts determine the relevancy of a threat for the system. An example threat tree is presented in Figure 2.2. Each path of the threat tree indicate a valid attack path. Note that some trees cascade.

For example, the tree in Figure 2.2 shows the conditions that could lead to tampering threats against a process. The node indicated as a circle (or oval) in the threat tree means a root threat, that cascades to another category. The node indicated as a rectangle suggest a concrete threat in an attack path. The arrows connecting the nodes in general refer to a *OR* relation among the various preconditions, unless it is indicated explicitly with "AND" to refer to a *AND* relation.

Afterwards, the identified privacy threats that are relevant to the designated system are documented as misuse cases, as a collection of threat scenarios in the system.

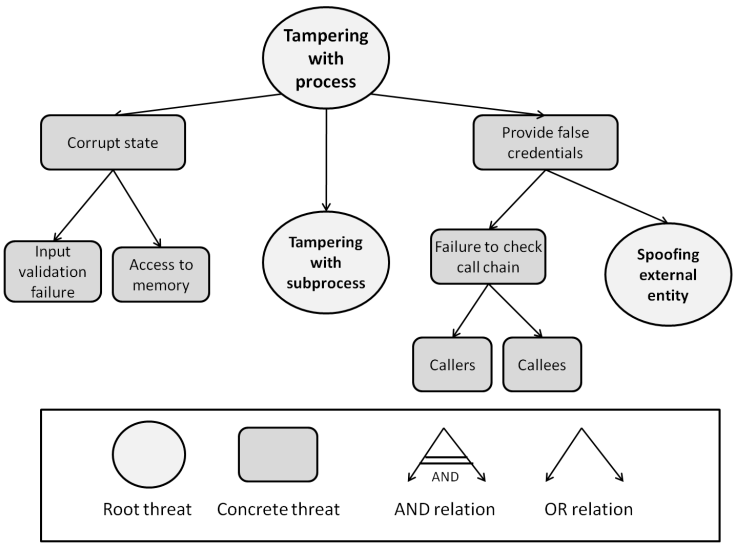


Figure 2.2. Example security threat tree pattern of tampering a process

8. *Determine risk.* For each threat, SDL has determined the appropriate security risk level, which can be used to define the priorities of the threats to be resolved.

9. *Plan mitigation.* In the final step of the methodology, the risk of the threat is reduced or eliminated by introducing proper countermeasures and defenses. Mitigating a risk to the threat corresponds to eliminating one attack path in the threat tree. An overview of some possible mitigation technologies linked to each security property is provided.

2.2.3 Security Threat Modeling Techniques

STRIDE comes with the main advantage of an extensive, reusable knowledge base (i.e. the threat tree patterns). However, some alternatives to elicit security threats exist.

Attack trees are similar to fault trees [264]. The root node describes the high-level attack which is further decomposed in lower-level attack branches. Each node represents a step that must be successfully executed in order to complete the attack represented by the parent node. Nodes can be composed in conjunctions and disjunctions. Attack trees can have both a graphical and a textual representation.

Misuse cases [51] or abuse cases are similar to regular use cases, however with a focus on the attacker's actions. Misuse cases have a textual representation, similar to use cases, and can also be represented in a misuse case diagram, which summarizes all existing misuse cases for a certain system and their impact on the system's use cases. Both techniques can be used to elicit security threats; these techniques however do not provide methodological guidance to discover additional threats. Opdahl and Sindre [31] have made an experimental comparison between attack trees and misuse cases of which the main finding was that attack trees are more effective for finding threats. Attack trees encourage the use of standard textbook threats and decomposition in lower-level threats. Misuse case analysis focuses more on user-level and organizational threats.

KAOS [284], a goal-oriented requirements analysis framework, has been extended with anti-goals to support the modeling of threats. Such anti-goals express the goals of an attacker who tries to abuse the system. Although no actual methodology exists to determine the threats, the root anti-goals are created by negating all the positive system goals. Next, the anti-goals are refined into trees. The formal nature of KAOS is an advantage which makes it possible to determine completeness of the (anti-)goals.

2.3 Our Approach – the LINDDUN Methodology

In this chapter, we propose a systemic approach for privacy threat modeling – the LINDDUN methodology – to elicit the privacy requirements and select privacy enhancing technologies accordingly. Each letter of "LINDDUN" stands for a privacy threat obtained by negating a privacy property. Privacy properties and threats are briefly described in Sections 2.4 and 2.5, respectively. In the STRIDE framework for security, explained in Section 2.2, the security countermeasures are directly proposed after the risk assessment. Therefore, STRIDE doesn't cover the elicitation of security requirements. We follow a slightly different approach for privacy, with an emphasis on privacy requirements elicitation.

The building blocks of the proposed LINDDUN methodology for privacy are depicted in Figure 2.3. In the figure, a distinction is marked between the proposed methodology and the supporting knowledge provided to assist each step. First of all, a data flow diagram is created based on the high-level system description. This is followed by mapping privacy threats to the DFD elements using Table 2.4 as a guide to determine the corresponding threats. In particular, a number of privacy tree patterns from Section 2.6 will be proposed to detail the privacy threat instances in a designated system, by providing an overview of the most common preconditions of each threat. Next, the identified privacy threats that are relevant to the designated system are documented as misuse cases in Section 2.7. A misuse case presents a collection of threat scenarios in the system. The identified privacy threats that needs to be evaluated and prioritized via risk assessment. Indeed, due to both time and budget constraints, not all threats are worthy further treatment. Note that details on the risk-analysis process are beyond the scope of this work. Hereafter, the privacy requirements of the system are elicited from the misuse cases following the mapping in Table 2.6. Finally, appropriate privacy enhancing solutions are selected according to the privacy requirements. Table 2.7 provides an overview of the state-of-art privacy enhancing techniques and the mapping to their corresponding privacy objectives.

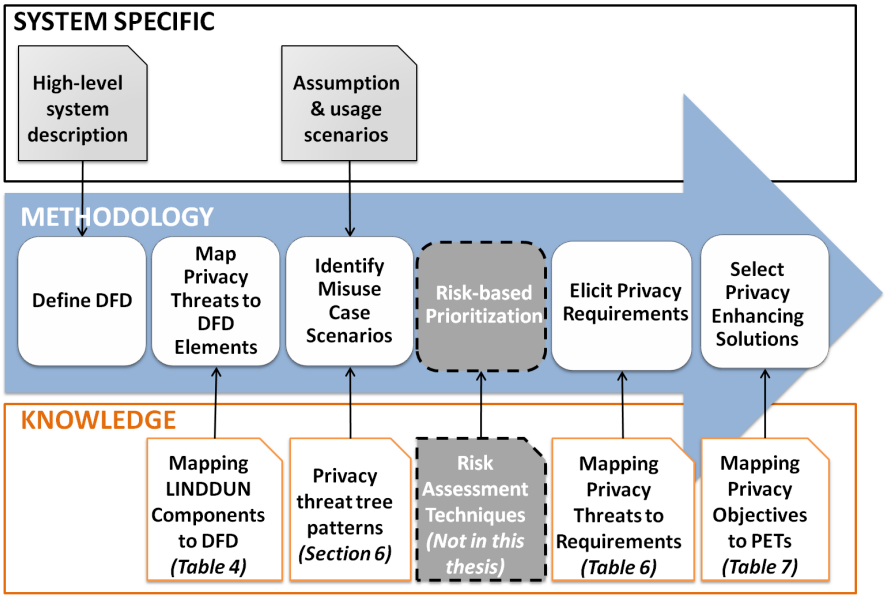


Figure 2.3. The LINDDUN methodology and the required system-specific knowledge

The fact that the LINDDUN framework and STRIDE are based on similar approaches creates synergy. Therefore, the privacy and security analysis can be nicely integrated into SDL, as shown in Figure 2.4. In bold, the nine steps of the security modeling process are enhanced with privacy-specific activities. Nevertheless, the aforementioned LINDDUN framework for privacy can be performed independently from the SDL security threat modeling process. In particular, privacy assumptions are specified (step 3), and external privacy notes are considered (step 4). This chapter proposes privacy-specific extensions to the key steps: determining privacy threat types (step 6) in Section 2.5.1 and identifying privacy threats (step 7) in Sections 2.5.2 to 2.7. Some initial suggestions for selecting feasible mitigation strategies towards privacy enhancing solutions are discussed in Section 2.8. Some initial ideas for step 8 *privacy risk assessment* are given in Section 2.7.1. However, this aspect is outside the scope for this thesis.

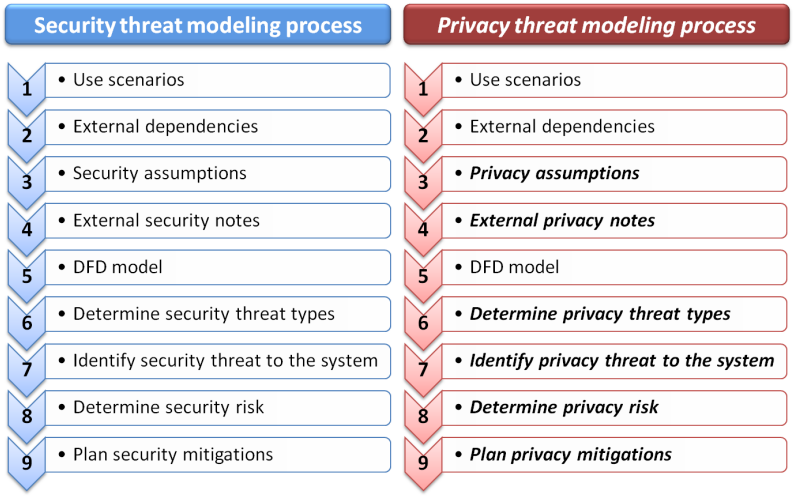


Figure 2.4. The integration of LINDDUN privacy threat modeling approach into the SDL threat modeling process

2.4 Privacy Properties

It is not the intention to propose a new taxonomy of privacy definitions in this chapter. However, it is crucial to have the right basis for the proposed LINDDUN framework, therefore definitions of privacy properties are elaborately studied and reviewed in this section. The literature is rich in studies to conceptualize privacy, and we suggest interested readers to refer to the work by Solove [272, 274]

for a comprehensive understanding of privacy. Most privacy properties in the LINDDUN framework comply with the terminology proposed by Pfitzmann et al. [237], as is widely recognized in the privacy research community.

2.4.1 Understanding Privacy: Hard Privacy Vs. Soft Privacy

The concept of privacy is introduced in Section 1.1.3. This section further elaborates on the privacy definitions. Privacy can be distinguished as *hard privacy* and *soft privacy*, as proposed by Danezis [109, 111]. The data protection goal of hard privacy refers to *data minimization*, or, protection against surveillance, interrogation, aggregation, and identification, using the taxonomic terms from Solove [272]. The system model of hard privacy is that a data subject, being an active security user, provides as little data as possible and tries to reduce the need to "trust" other entities as much as possible, and it also implies the assumption that personal data is not divulged to third parties. The threat model includes communication and service provider, data holder, and adversarial environment, where strategic adversaries with certain resources are motivated to breach privacy, similar to security systems.

Soft privacy, on the contrary, refers to the assumption that the data subject has already lost control of his or her data and needs to trust the honesty and competence of data controllers. This concept is motivated because it is difficult for the data subject to verify how his or her data is collected and processed in practice, for example, there are millions of exposed records per year due to data breaches at businesses, government agencies and other institutions. In short, the data protection goal of soft privacy is to provide data security and process data with specific purpose and consent, by means of policies, access control, and audit. In the system model of soft privacy, the data subject provides his or her data, and data controller, as the main security user, is responsible for the data protection. Unfortunately, it brings the technical disadvantage of soft privacy solutions because there is no direct control from the data subject and it is impossible to defend against a malicious data holder or super-user. Consequently, a weaker threat model applies including different parities with inequality of power, such as external parties, honest insiders who make errors, and corrupt insiders within honest data holders. An overview of hard and soft privacy solutions will be given in Section 2.8.

It is argued that privacy should become a first-class security property, suggested by Danezis [110] for a number of reasons. Primarily, the concept of self-determination is considered as the most valued security property. In addition, privacy satisfies valid security needs of some entities, such as freedom from surveillance and profiling, freedom from compulsion, and exibility to access and use content and services. Furthermore, comparable with security, privacy needs

to be technologically supported and laws are necessary but insufficient to protect privacy.

Besides conceptualizing privacy, another research challenge is to define privacy properties. Since secure communication channel is one of the basic requirements for privacy, some classical security properties are desired as privacy properties, including *confidentiality* (ensuring that information is accessible only to those authorized to have access), *integrity* (safeguarding the accuracy and completeness of information and processing methods), *availability* (also censorship resistance, information is accessible to authorized users), and *non-repudiation* (one should not be able to deny what one has done). The official definitions of these properties can be found in ISO 17799 [10].

In addition, a number of properties are also appreciated, including *anonymity* (hiding links between identity and action or a piece of information), *unlinkability* (hiding link between two or more actions, identities and pieces of information), *undetectability* (or covertness) and *unobservability* (hiding user's activity), *plausible deniability* (opposite of non-repudiation, no others can prove one has said or done something), and *forward security* (also referred as forward secrecy and freedom from compulsion, meaning that once the communication is securely over, it cannot be decrypted any more).

We decided to include the following privacy properties in the proposed framework, namely unlinkability, anonymity and pseudonymity, plausible deniability, undetectability and unobservability, and confidentiality (hiding data content, including access control) as hard privacy properties; *user content awareness* (including feedback for user privacy awareness, data update and expire) together with *policy and consent compliance* as soft privacy properties. These properties are described in the following sections. Note that properties such as integrity, availability, and forward security are also important for privacy. However, we consider them as typical security properties; hence they are to be considered in the security engineering framework, such as STRIDE.

2.4.2 Unlinkability

The unlinkability property refers to hiding the link between two or more actions, identities, and pieces of information. Examples of unlinkability include hiding links between two anonymous messages sent by the same person, two web page visits by the same user, entries in two databases related to the same person, or two people related by a friendship link in a social network.

Unlinkability is defined by ISO 15408 [3] as: *"Unlinkability ensures that a user may make multiple uses of resources or services without others being able to link these uses together. [...] Unlinkability requires that users and/or subjects are*

unable to determine whether the same user caused certain specific operations in the system [3]."

The above definition focuses on linking processes (i.e., uses of resources or services) by the same users. To extend this, the definition from Pfitzmann et al. [237] focuses on linking objectives (IOIs), and hence refers to the DFD elements of entity, data flow, and data store. *"Unlinkability of two or more items of interest (IOIs, e.g., subjects, messages, actions, ...) from an attacker's perspective means that within the system (comprising these and possibly other items), the attacker cannot sufficiently distinguish whether these IOIs are related or not [237]."*

When referring to comparing whether two or more IOIs are linked, the aforementioned definition of unlinkability suggests implicitly that these IOIs are of the same DFD element type. Although it is not explicitly mentioned, the definition of unlinkability implies that the two or more IOIs are of the comparable types, in order to facilitate the comparison of these IOIs.

2.4.3 Anonymity

Essentially, the anonymity property refers to hiding the link between an identity and an action or a piece of information. Examples are anonymous sender of an email, writer of a text, person accessing a service, person to whom an entry in a database relates, and so on.

Pfitzmann et al. [237] pointed out that to enable anonymity of a subject, there always has to be an appropriate set of subjects with potentially the same attributes, the anonymity set. This leads to the first definition: *Anonymity of a subject means that the subject is not identifiable within a set of subjects, the anonymity set [237].*

The definition given above basically defines anonymity as a binary property: either a subject is anonymous or not. It emphasizes one entity and an entire anonymity set. To reflect the possibility to quantify anonymity and to underline that all statements are made from the perspective of an attacker, the definition of anonymity can be extended as: *"Anonymity of a subject from an attacker's perspective means that the attacker cannot sufficiently identify the subject within a set of subjects, the anonymity set." [237]*

Anonymity can also be described in terms of unlinkability. If one considers sending and receiving of messages as attributes; the items of interest (IOIs) are those who have sent or received which message. Then, *"anonymity of a subject with respect to an attribute may be defined as unlinkability of this subject and this attribute."* For instance, *sender anonymity* of a subject means that to this potentially sending subject, each message is unlinkable. Correspondingly, *recipient anonymity* of a subject means that to this potentially receiving subject, each message is unlinkable.

2.4.4 Pseudonymity

The pseudonymity property suggests that it is possible to build a reputation on a pseudonym and possible to use multiple pseudonyms for different purposes. Examples include a person publishes comments on social network sites under different pseudonyms and a person uses a pseudonym to subscribe to a service.

Pfitzmann et al. [237] defines pseudonymity as: *"A pseudonym is an identifier of a subject other than one of the subject's real names. Pseudonymity is the use of pseudonyms as identifiers. A subject is pseudonymous if a pseudonym is used as identifier instead of one of its real names."* Pseudonymity can also be perceived with respect to (un)linkability. Whereas anonymity and identifiability (or accountability) are the two extremes with respect to (un)linkability to subjects, pseudonymity is the entire field between and including these extremes. Therefore, pseudonymity comprises all degrees of (un)linkability to a subjective's identity.

2.4.5 Plausible Deniability

For privacy, plausible deniability refers to the ability to deny having performed an action that other parties can neither confirm nor contradict. Plausible deniability from an attacker's perspective means that an attacker cannot prove a user knows, has done or has said something. Sometimes, depending on the application, plausible deniability is desirable over non-repudiation, for instance, in an application used by whistleblowers, users will want to deny ever having sent a certain message to protect their safety. Other examples include off-the-record conversations, possibility of denying the existence of an encrypted file, denying that a file is transmitted from a data source, or denying that a database record belongs to a person.

The relation between non-repudiation and plausible deniability is outlined by Roe [261]: *"The goal of the non-repudiation service is to provide irrefutable evidence concerning the occurrence or non-occurrence of an event or action. If we believe that there is a need for this as a security service[...] we must also concede that some participants desire the opposite effect: that there be no irrefutable evidence concerning a disputed event or action." This "complementary service" is plausible deniability.* "Non-repudiation and plausible deniability are mutually exclusive in that an entity can't both have and not have sufficient evidence to convince a particular party that a particular event happened. However, it is possible to imagine combinations of the two services in which some parties retain evidence of an event while others don't, or the evidence is sufficient to convince some parties but not others [261]."

In particular, it ensures that "an instance of communication between computer systems leaves behind no unequivocal evidence of its having taken place. Features

of communications protocols that were seen as defects from the standpoint of non-repudiation can be seen as benefits from the standpoint of this converse problem, which is called plausible deniability."

Mao et al. described plausible deniability in a computer system: *"the plausible deniability property is meant to ensure that an entity A's involvement in a protocol run with another entity B does not generate any evidence that can be used to demonstrate (either by B or by some third party who observes the protocol run) that A and B did participate in a protocol run. This property[...] is usually achieved in such authenticated protocols by avoiding the direct signing of identities in exchanged messages [200]."*

2.4.6 Undetectability and Unobservability

The undetectability and unobservability properties refer to hiding the user's activities. Practical examples include that, it is impossible to know whether an entry in a database corresponds to a real person, or to distinguish whether someone or no one is in a given location.

ISO 15408 [3] defines unobservability as: "*Unobservability ensures that a user may use a resource or service without others, especially third parties, being able to observe that the resource or service is being used. [...] Unobservability requires that users and/or subjects cannot determine whether an operation is being performed.*" Essentially, this definition refers an entity (subject) to data flows (e.g. a messages has been sent), data stores (e.g. a database has been accessed), and processes (a resource or service has been used).

To be precise, Pfitzmann et al. make the distinction of undetectability from unobservability. "*Undetectability of an item of interest (IOI) from an attacker's perspective means that the attacker cannot sufficiently distinguish whether it exists or not. If we consider messages as IOIs, this means that messages are not sufficiently discernible from, for example, random noise [237].*" While the anonymity and unlinkability properties require the protection of, instead of an IOI itself, the relationship between the IOI to a subjects or other IOIs, the undetectability property requires the protection of IOIs.

Undetectability of an IOI against uninvolved subjects and, at the same time, anonymity of involved subjects even when IOIs can be detected, is defined as unobservability [237]: "*Unobservability of an item of interest (IOI) means undetectability of the IOI against all subjects uninvolved in it and anonymity of the subject(s) involved in the IOI even against the other subject(s) involved in that IOI.*" The definition suggests that unobservability is undetectability by uninvolved subjects AND anonymity even if IOIs can be detected. Consequently, unobservability implies anonymity, and unobservability implies undetectability. It

means, with respect to the same attacker, unobservability reveals always only a subset of the information anonymity reveals. Later sections of this chapter will focus on undetectability, since unobservability is in fact a combination of undetectability and anonymity.

2.4.7 Confidentiality

The confidentiality property refers to hiding the data content or controlled release of data content. Examples include transferring encrypted email, applying access control to a classified document or a database containing sensitive information.

NIST [201] describes confidentiality as followings: *Confidentiality means preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information.* Although confidentiality is a security property, as the definition above states, it is also important for preserving privacy properties, such as anonymity and unlinkability. Therefore, confidentiality is also considered an important privacy property.

Not only crucial for security, confidentiality is, as the definition above states, also significant for privacy. Several privacy objectives, such as anonymity and unlinkability, depend on confidentiality. For instance, in order to keep the user's identity hidden, the communication between the user and the system needs to be confidential. Therefore, confidentiality is also considered an important privacy objective.

2.4.8 Content Awareness

Unlike the aforesaid classical privacy properties, to our knowledge, the following two properties, namely content awareness, and policy and consent compliance, are not explicitly defined in the literature. However, we consider them important privacy objectives, due to their significance to privacy and data protection. With the emerging of Web 2.0 technologies, users tend to provide excessive information to service providers and lose control of their personal information. Therefore, the content awareness property is proposed to make sure that users are aware of their personal data and that only the minimum necessary information should be sought and used to allow for the performance of the function to which it relates.

The more personal identifiable information a data subject discloses, the higher the risk is for privacy violation. To ensure content awareness, a number of technical enforcement tools have been developed. For instance, the concept of personal information feedback tools has been promoted [188, 231] to help users gain privacy awareness and self-determine which personal data to disclose.

The Platform for Privacy Preferences Project (P3P) [229] has been designed to allow websites (as data controllers) to declare their intended use of the information that they collected about the browsing users (as data subjects). P3P addresses the content awareness property by making users aware of how personal data are processed by the data controller.

Although not necessarily privacy-oriented, within the realm of content awareness objective, another responsibility of the user is to keep user's data up-to-date to prevent wrong decisions based on incorrect data. This means that the data subject or the data controller (depends on applications) is responsible for deleting and updating inaccurate information. For example, it is crucial to maintain patient's data in e-health applications. Imagine a doctor forgetting to mention that the patient is a diabetic, the absence of information could cause fatal consequences for patients taking medication without considering negative side effects on diabetics.

To summarize, the content awareness property focuses on the user's consciousness regarding his own data. The user needs to be aware of the consequences of sharing information. These consequences can refer to the user's privacy, which can be violated by sharing too much personal identifiable information, as well as to undesirable results by providing incomplete or incorrect information.

2.4.9 Policy and Consent Compliance

Unlike the content awareness property focused on the user, the policy and consent compliance property requires the whole system – including data flows, data stores, and processes – as data controller to inform the data subject about the system's privacy policy, or allow the data subject to specify consents in compliance with legislation, before users accessing the system. According to the definitions from the EU Directive 95/46/EC [140]: *"Controller shall mean the natural or legal person, public authority, agency or any other body which alone or jointly with others determines the purposes and means of the processing of personal data." "The data subject's consent shall mean any freely given specific and informed indication of his wishes by which the data subject signifies his agreement to personal data relating to him being processed."*

A policy specifies one or more rules with respect to data protection. These are general rules determined by the stakeholders of the system. Consents specify one or more data protection rules as well, however, these rules are determined by the user and only relate to the data regarding this specific user. The policy and consent compliance property essentially ensures that the system's policy and the user's consent, specified in textual form, are indeed implemented and enforced.

This property is closely related to legislation. There are a number of legal frameworks addressing the raised concerns of data protection, such as the Health

issued Insurance Portability and Accountability Act (HIPAA) [16] in the United States, the Data Protection Directive 95/46/EC [140] in Europe, the Personal Information Protection and Electronic Documents Act and Privacy Act [34] in Canada, the Commonwealth Privacy Act 1988 and Privacy Amendment (Private Sector) Act 2000 [216] in Australia, and the OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data [223].

One example of consent compliance is in e-health, for some countries, healthcare professionals are not allowed to intervene until the data subject has given informed consent for medical treatment.

There are initiatives to protect data subjects and create openness; however it is evidently important to ensure that internal rules actually comply with that promised in policies and consents. Unfortunately, few technical solutions exist to guarantee the compliance. A possible non-technical solution is to use employee contracts to enforce penalties (e.g., get fired or pay fines) to ensure compliance. Another solution is to hire an auditor to check policies compliance. Eventually, necessary legal actions can be taken by data subjects in case of noncompliance.

Breaux et al. [79] pointed out that to ensure a product that complies with its privacy and security goals, legal requirements need to be identified and refined into product requirements, and the product requirements need to be integrated into the ongoing product design and testing processes. They presented an industry case study in which requirements of Cisco products were specified to comply with Section 508 of the U.S. Workforce Investment Act (WIA) of 1998 [224]. They developed a set of qualitative metrics to rationalize the comparison of two requirements. These metrics demonstrate that alignments between legal and product requirements can be described in detail by using the goal-oriented concept of refinement. Their analysis revealed that a frame-based requirements analysis method [78], which itemizes requirements and preserves legal language, is useful to incorporate legal requirements into a manufacturer's compliance framework.

2.5 Mapping Privacy Threats to DFD

In this section, we present the privacy threat categories based on the aforementioned privacy properties. We also discuss how to map these categories to the DFD elements.

2.5.1 Privacy Threat Categories

As shown in Table 2.2, the methodology considers seven types of threats. LINDDUN is the mnemonic acronym that we use.

Table 2.2. In the LINDDUN methodology, privacy properties and the corresponding privacy threat are categorized as hard privacy and soft privacy

	Privacy properties	Privacy threats
HARD	Unlinkability	**L**inkability
	Anonymity & Pseudonymity	**I**dentifiability
	Plausible deniability	**N**on-repudiation
	Undetectability & Unobservability	**D**etectability
	Confidentiality	**D**isclosure of information
SOFT	Content awareness	content **U**nawareness
	Policy and consent compliance	policy and consent **N**oncompliance

The following section describes LINDDUN components:

1. *Linkability* of two or more items of interest (IOIs, e.g., subjects, messages, actions, etc.) allows an attacker to sufficiently distinguish whether these IOIs are related or not within the system.

2. *Identifiability* of a subject means that the attacker can sufficiently identify the subject associated to an IOI, for instance, the sender of a message. Usually, identifiability refers to a set of potential subjects, called the identifiability set [237]. In essence, identifiability is a special case of linkability when a subject and its attributes are involved. Identifiability is a threat to both anonymity and pseudonymity.

3. *Non-repudiation*, in contrast to security, this is a threat for privacy. Non-repudiation allows an attacker to gather evidence to counter the claims of the repudiating party, and to prove that a user knows, has done or has said something.

4. *Detectability* of an IOI means that the attacker can sufficiently distinguish whether such an item exists or not. If we consider messages as IOIs, it means that messages are sufficiently discernible from random noise.

5. *Information Disclosure* threats expose personal information to individuals who are not suppose to have access to it.

6. *Content Unawareness* indicates that a user is unaware of the information disclosed to the system. The user either provides too much information which allows an attacker to easily retrieve the user's identity or inaccurate information which can cause wrong decisions or actions.

7. *Policy and consent Noncompliance* means that even though the system shows its privacy policies to its users, there is no guarantee that the system actually complies to the advertised policies. Therefore, the user's personal data might still be revealed.

2.5.2 Mapping Privacy Threat Categories to The System

This section provides an guideline to identify privacy threats of an application system. First, a Data Flow Diagram (DFD) is created in correspondence to the application's use case scenarios. Second, privacy threats are mapped to the DFD.

Creating Application DFD Based On Use Case Scenarios

DFD is chosen to present a system based on two reasons. First, DFD is proved to be sufficiently expressive in a number of case studies examined by the authors. Second, DFD is also used by the SDL threat modeling process, hence by deploying the same modeling technique an interesting synergy can be created between the proposed framework and the SDL process.

Running example: Social Network 2.0

In our running example Social Network 2.0, Alice is a registered user of a social network. Each time Alice updates her friends list, she first connects to the social network's web portal. Accordingly, the portal communicates with the social network's server, and eventually, the friendship information of Alice and all other users of that social network is stored in a database.

The DFD for the Social Network 2.0 application was already presented in Figure 2.1 of Section 2.2. Table 2.3 lists the DFD elements.

The creation of the DFD is an important part in the analysis. If the DFD was incorrect, the analysis results would be wrong as well. Since privacy focusses on the protection of user's personal information, it is important to consider where the information will be stored or passed by, as these are the crucial elements for building in privacy.

Mapping Privacy Threats to DFD

After the DFD elements are listed, we identify the privacy threat categories for each DFD element by following the mapping depicted in Table 2.4. Each intersection marked with the symbol $\times$ indicates a potential privacy threat at a corresponding DFD element in the system.

Table 2.3. DFD elements in the Social Network 2.0 application

Entity	User
Process	Portal Social network service
Data Store	Social network database
Data Flow	User data stream (user – portal) Service data stream (portal – service) Database data stream (service – database)

Table 2.4. Mapping LINDDUN components (privacy threats) to DFD element types (E-Entity, DF-Data flow, DS-Data store, P-Process)

Threat categories	E	DF	DS	P
Linkability	×	×	×	×
Identifiability	×	×	×	×
Non-repudiation		×	×	×
Detectability		×	×	×
Information **D**isclosure		×	×	×
Content **U**nawareness	×			
Policy/consent **N**oncompliance		×	×	×

In essence, each DFD element is subject to certain privacy threats, and the nature of the potential privacy threat is determined by the DFD element type. For example, a data flow is subject to a number of privacy threats such as identifiability, linkability, detectability, non-repudiation, and information disclosure. The following sections will explain how privacy threats affect DFD elements. More threat scenarios corresponding to our running example will be discussed in Section 2.7.

The nature of *linkability* indicates that the threat affects DFD elements pairwise. In other words, linkability of a DFD element refers to a pair (x_1, x_2), where $x_i \in \{E, DF, DS, P\}$ is the linkable IOI. Obviously, linkability at entity, from an attacker's perspective means that within the system (comprising these and possibly other items), the attacker can sufficiently distinguish whether these entities are related or not. Similar description applies for that of data flow, data store, and process.

The *identifiability* threat affects all four DFD elements, such that each DFD element is made explicit as the attributes that identifiability (or its opposite

property anonymity) relates to, by forming a pair with a subject. Essentially, identifiability at each DFD element refers to a pair (x, y), where $x \in \{E\}$ is the identifiable subject, and $y \in \{E, DS, DF, P\}$ is the attribute identifiability relates to. For example, identifiability at entity refers to a pair (E, E), meaning to identify an entity within a set of entities. Identifiability at data flow refers to a pair (E, DF), meaning that a message is linkable to a potentially sending or receiving subject. Identifiability at data store refers to a pair (E, DS), meaning that a database entry is linkable to a potential data holder or subject. Identifiability at process refers to a pair (E, P), meaning that a process is linkable to a potentially accessing subject.

Non-repudiation, opposite of plausible deniability, is a privacy threat that affects the DFD elements of data flow, data store and process. Non-repudiation might be appreciated for some system but undesirable for others. It depends on the system requirements. For e-commerce applications, non-repudiation is an important security property. Imagine a situation where a buyer signs for a purchased item upon receipt, the vendor can later use the signed receipt as evidence that the user received the item. For other applications, such as off-the-record conversations, participants may desire plausible deniability for privacy protection such that there will be no record to demonstrate the communication event, the participants and the content. In this scenario, non-repudiation is a privacy threat. Even though entity is the only DFD element being able to (non-)repudiate, the non-repudiation privacy threat actually occurs at data flow, data store, and process. Similar to linkability and identifiability, non-repudiation at each DFD element refers to a pair (x, y), where $x \in \{E\}$ is the non-repudiating subject, and $y \in \{DS, DF, P\}$ is the attribute it relates to.

Detectability threats occur at data flow, data store, and process, meaning that the attacker can sufficiently distinguish whether it exists or not. Though in some applications, techniques such as covert channel and steganography can be used to protect both messages (data flow) and communicating parties (entity), in this case the threat actually occurs at data flow instead of entity. In other words, the asset we want to protect against the detectability threat includes data flow, data store, and process.

Information disclosure threats affect data flow, data store, and process, referring to the exposure of information at these DFD elements to individuals who are not supposed to have access to it.

The *content unawareness* threat is related to entity, since the entity (data subject or data controller) is actually responsible to provide the necessary consents to process personal data and update or delete the expired information.

Policy and consent noncompliance is a threat that affects system as a whole, because each system component (including data flow, data store and process) is responsible to ensure that actions are taken in compliance with privacy policies

Table 2.5. Determining privacy threats (LINDDUN components) to DFD elements within the Social Network 2.0 application (From left to right: L-Linkability, I-Identifiability, N-Non Repudiation, D-Detectability, D-Information Disclosure, U-Content Unawareness, N-Consent/policy Noncompliance)

	Threat target	L	I	N	D	D	U	N
Data Store	Social network database	1	4	×	×	7		10
Data Flow	User data stream	2	5	×	×	8		10*
	Service data stream	×	×	×	×	×		10*
	Database data stream	×	×	×	×	×		10*
Process	Portal	×	×	×	×	×		10*
	Social network service	×	×	×	×	×		10*
Entity	User	3	6				9	

and data subject's consents.

Running example: Social Network 2.0

Considering the Social Network 2.0 application, the list of generic privacy threats to the modeled system is depicted in Table 2.5. This is obtained by gathering the elements from Table 2.3 and then determining the susceptible threats using Table 2.4.

The intersections marked with × in Table 2.5 are potential threats that have been considered as irrelevant to the specific usage scenario. Each intersections that is indicated with a number (1 to 10) in Table 2.5 means that there will be a privacy threat at the corresponding DFD element. These items marked with a number are the threats which we will actually consider. The number represents the ID of the generic threat and will be used later for ease of reference.

Primarily, we assume that DFD elements within the trust boundary (marked as dashed line in Figure 2.1) are trustworthy. We trust the processes with the boundary, as well as all data flows in the trust boundary. Therefore, we won't discuss linkability, identifiability, and information disclosure threats on these elements. We however do not trust the user and its communication with the portal and we also want to protect the data store containing all the user's information.

Moreover, non-repudiation and detectability threats are considered irrelevant for social networks. Presumably, it depends on what privacy properties are required for a particular social network system. In case plausible deniability and undetectability would be desirable for a certain application, we should still consider these threats to each DFD element accordingly.

Following the above reasoning, ten threats will be examined in detail in Section 2.7, and they are numbered in Table 2.5. Note that some items are indicated with a $10*$. This means that the policy and consent noncompliance threat affects the system as a whole (including data flow, data store and process).

2.6 Detailing privacy threats via threat tree patterns

This section presents a (significant) number of threat tree patterns that can be used to detail the privacy threats to a system. In the STRIDE framework for security, which was explained in Section 2.2, the security countermeasures are directly proposed after the risk assessment. STRIDE doesn't cover the elicitation of security requirements. A different approach is proposed for privacy. The goal is to use privacy threat trees to elicit privacy requirements. This can be achieved in a number of steps. First, the threat tree patterns are used as an indication to allow system analysts to consider the currently most common privacy conditions and specific threat instances that can occur in a system. This step is followed by a selection process through risk assessment, in which less relevant threat instances are to be ranked with a lower priority and discarded. Details on the risk-analysis process are beyond the scope of this chapter. In the third step, the identified privacy threats that are relevant to the designated system are documented as misuse cases (to be explained in Section 2.7), which assemble a collection of threat scenarios in the system. Finally, privacy requirements of the system will be elicited from the documented misuse cases (to be explained in Section 2.8.1).

The privacy threat trees are inspired by the Secure Development Lifecycle (SDL) and based on the state-of-art privacy developments. These threat trees reflect common attack patterns and help application designers think about privacy conditions in the system. This section itemizes the privacy threat tree patterns and discusses what the designers should consider when designing and testing an application.

This is assumed that the ideal pattern library is sufficiently large to ensure that all possible violations will be covered. The threat trees depicted in this section present our best effort so far. Further completeness of the proposed the trees is subject to validations by practitioners for applications in real life systems. Comparable to the SDL threat tree patterns, the privacy threat trees are susceptible to a continuous improvement process, based on both the existing and the newly discovered threats.

For each marked intersection in Table 2.4, a threat tree exists showing the detailed preconditions for this specific threat category to materialize. The preconditions are hence vulnerabilities that can be exploited for a privacy attack scenario.

2.6.1 Linkability of Entity

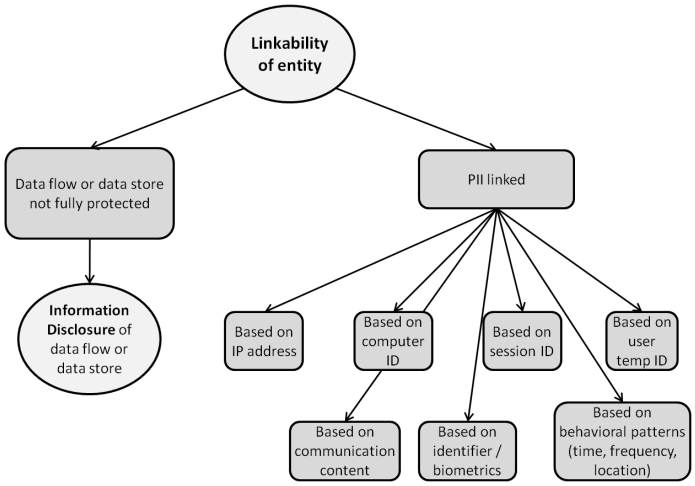


Figure 2.5. Threat tree for linkability of an entity

Linkability of entity refers to when an attacker can sufficiently distinguish whether two or more entities are related or not within the system. This implies that different pseudonyms can be linked to each other. The threat tree pattern is depicted in Figure 2.5. One precondition is that data flow or data store is not fully protected (e.g. unencrypted), which leads to the Information Disclosure threat of data flow and data store. The second precondition is that Personal Identifiable Information (PII) can be linked, e.g. based on user temporary ID, IP address, behavioral patterns such as time, frequency and location, session ID, identifier and biometrics, computer ID, communication content or any combination of these factors. The aforementioned data store refers to the identity management system's database or any other database which contains personal identifiers of users. Having accessed such a data store, the attacker could easily link different pseudonyms to the same user.

2.6.2 Linkability of Data Flow

The *linkability of data flow* threat tree, as presented in Figure 2.6, suggests two preconditions. One precondition is that data flows are not fully protected (e.g. unencrypted), which leads to information disclosure of data flow; and communications are linkable due to little or insecure anonymity systems deployed. The other precondition is that communication can be linked. When no anonymous

communication is deployed, basically the same preconditions apply as for the linkability of entity threat. Messages are linked to each other by user's identifiable information (e.g. based on user temporary ID, IP address, behavioral patterns such as time, frequency and location, session ID, identifier and biometrics, computer ID, communication content or any combination of these factors). Alternatively, when an insecure anonymity system is deployed, traffic analysis is possible to extract information out of patterns of traffic; passive attacks (e.g. long-term intersection attacks, traffic correlation and confirmation, fingerprinting, epistemic attacks (route selection), and predecessor attacks) and active attacks (e.g. N-1 attacks, Sybil attack, traffic watermarking, tagging attack, replay, and DoS attack) are possible to link entities together. An overview of these attacks can be found in [112].

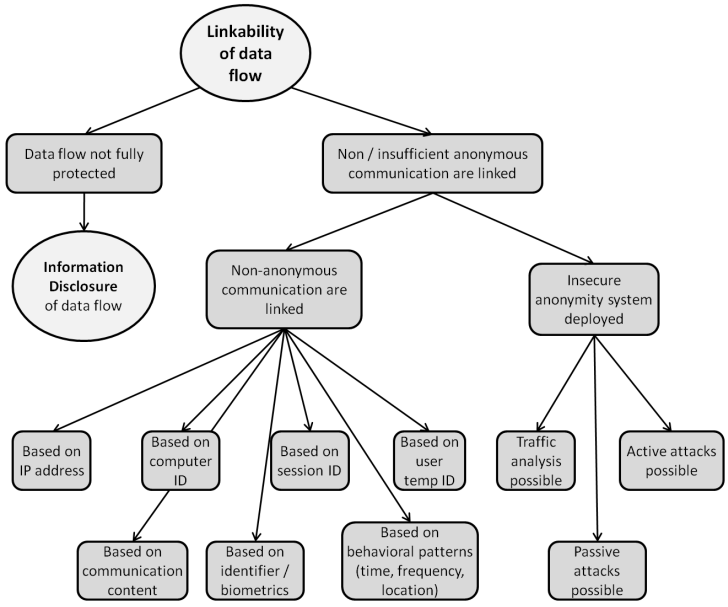


Figure 2.6. Threat tree for linkability of a data flow

2.6.3 Linkability of Data Store

Two preconditions correspond to the threat of *linkability of a data store*, as shown in Figure 2.7. First, there is insufficient access control of the data store leading to the information disclosure threat at a data store. Second, insufficient data anonymization is applied or strong data mining is possible in the data store, meaning that the stored information still contains sufficient references to the

corresponding data subject, which makes it possible to link different data items to each other within the same database. Another possibility is that data can be linked from one database to another, and hence re-identification [276] is possible.

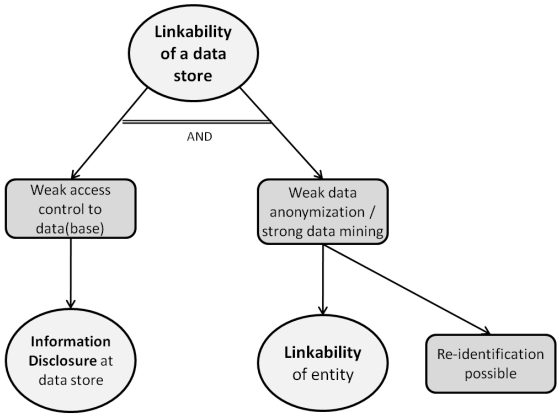


Figure 2.7. Threat tree for linkability of a data store

2.6.4 Linkability of Process

The threat tree of *linkability of process* suggests that the only way to prevent different actions being linked to the same subject is by gaining access to the process, as illustrated in Figure 2.8.

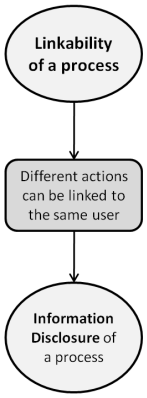


Figure 2.8. Threat tree for linkability of a process

2.6.5 Identifiability of Entity

Figure 2.9 shows the threat tree pattern for *identifiability of entity*. It gives an overview of the most common situations where an identifiability threat can occur at an entity. A first precondition is when the e-id is used as login (i.e., the user's actual identity is used), meanwhile the data flow between the user and login portal is not sufficiently protected (i.e., the threat of information disclosure of data flow), and thus the user's identity will be exposed. A second possibility occurs when a secret (e.g. a password) is used as log-in and the relationship between this secret and the user can be revealed. This can happen if the identity management database in not secure (e.g. passwords are stored in clear); if the communication channel is insecure and the communicated passwords are weak and can be connected to the user (e.g. using a birthdate as password); or if replay attacks are possible (e.g. a keylogger is installed, the communication can be eavesdropped or the person entering the secret can be observed). A third possible precondition for the threat is the use of a token as log-in which is weakly implemented or physically insecure. The final precondition is that biometrics is used as log-in, which means that biometrics is retrievable and can be linked to an entity. It is due to information disclosure of identity management database or data flow which contains biometrics, and linkability at data store.

2.6.6 Identifiability of Data Flow

The threat tree of *identifiability of data flow* is presented in Figure 2.10. Similar to the linkability of data flow threat tree, identifiability of data flow is possible when data flows are not fully protected, which leads to information disclosure of data flow; or when the communication can be traced to an entity due to little or insecure anonymity system deployed. When no anonymity system is deployed, communication can be traced to an entity by means of identifiable information (e.g. based on user temporary ID, IP address, behavioral patterns such as time, frequency and location, session ID, identifier and biometrics, computer ID, communication content or any combination of these factors). Alternatively, when an insecure anonymity system is deployed, traffic analysis, passive attacks, and active attacks [112] are possible to identify the particularly entity of interests.

2.6.7 Identifiability of Data Store

The preconditions for *identifiability of data store*, as presented in Figure 2.11, are similar to the preconditions of linkability at a data store. Either there is insufficient access control of the data store which refers to the information disclosure threat of a data store; or insufficient data anonymization or strong data mining techniques

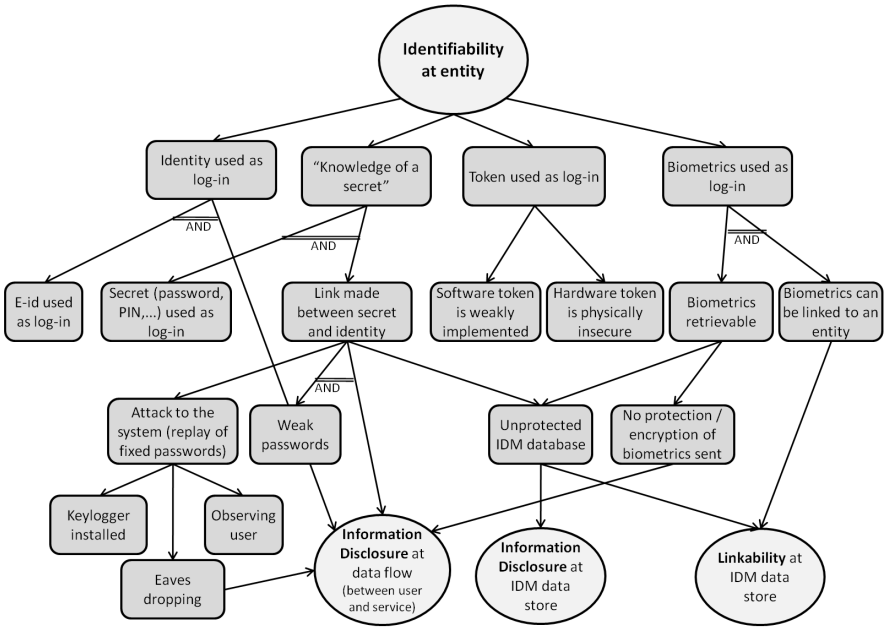


Figure 2.9. Threat tree for identifiability of an entity

where applied in the data store, which means that the information stored still contains sufficient references to the corresponding person to reveal the person's identity, and hence it refers to the identifiability threat of entity or the data can be linked with other relevant information which can lead to re-identification of the data subject.

2.6.8 Identifiability of Process

The threat tree of *identifiability of process* is presented in Figure 2.12. The only condition for the identifiability threat at a process is when access to the process is not sufficiently secured, and thus it refers to the threat of information disclosure of a process.

2.6.9 Non-repudiation of Data Flow

Four general preconditions can be applied to the threat tree of *non-repudiation of data flow*, as presented in Figure 2.13. One condition is insufficient obfuscation

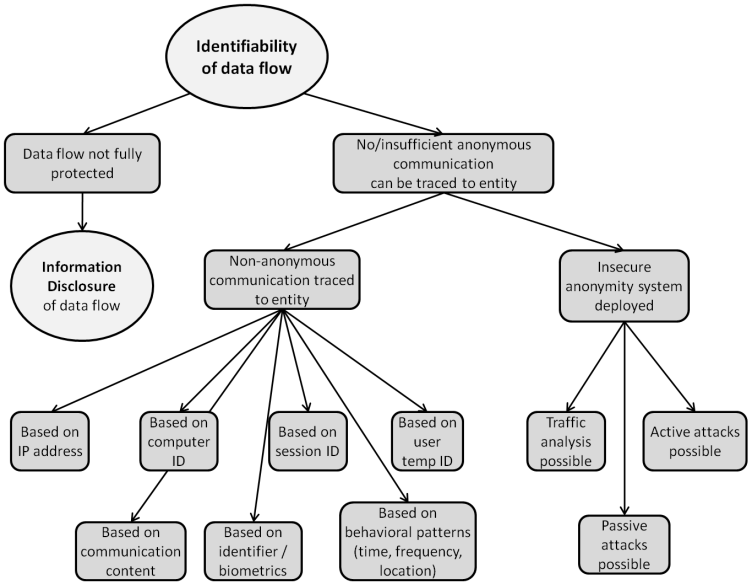


Figure 2.10. Threat tree for identifiability of a data flow

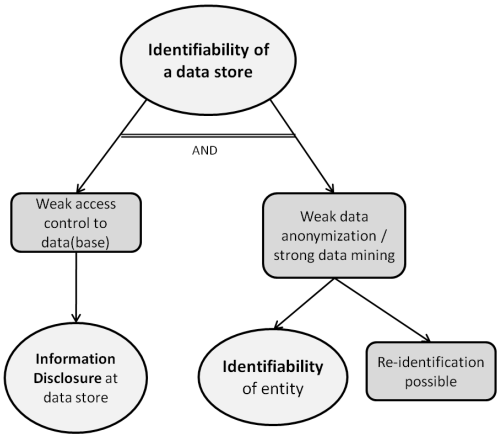


Figure 2.11. Threat tree for identifiability of a data store

for data sources or data flows, which means that the attacker can gain access to at least part of the data flow or data source. This can occur in a number of cases, for example, there is no automatic replay of broadcasts, such that the sender of a

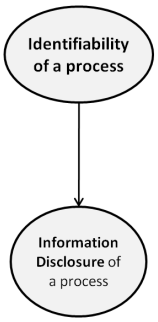


Figure 2.12. Threat tree for identifiability of a process

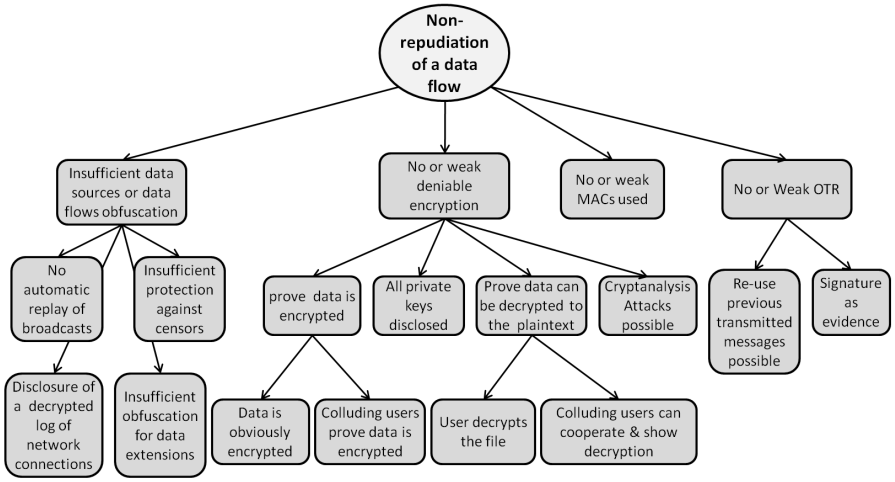


Figure 2.13. Threat tree for Non-repudiation of a data flow

file is sufficiently distinguishable from those who are merely relaying it. Another example is when a complete decrypted log of all network connections to and from a user's computer is disclosed, resulting in the disclosure of the origin of data flow. The final examples are that there is insufficient protection against censors or insufficient obfuscation of data extensions, such that operators or users of the network are able to know where the data comes from.

The second precondition of this threat is that little or a weak deniable encryption technique is used to protect data flow. One possible attack path is to prove data is encrypted, either due to the encrypter proves the data is obviously an encryption

or colluding users prove together that the data is encrypted. The second attack path is to prove data can be decrypted to a valid plain text, which can occur when the encrypter decrypts the file or colluding users can cooperate and show the decrypted message. The third attack path shows that all private keys are disclosed, and the last path suggests that cryptanalysis is possible to attack the used encryption scheme.

The third condition is that there are little or weak message authentication codes (MAC) used to ensure integrity of data flow content, such that an attacker can forge authentic looking messages and pretend that a certain data flow comes from a subject.

The final precondition indicates that there is little or a weak Off-the-Record Messaging (OTR) used, such that in a conversation it is not possible to provide both deniability for the conversation participants and confidentiality of conversations content at the same time. Possible attack paths include replaying of previous transferred messages, and the use of signatures to demonstrate communication events, participants and communication content.

2.6.10 Non-repudiation of Data Store

The threat tree for *non-repudiation of data store* is depicted in Figure 2.14. Three preconditions can apply to this threat, namely a weak access control to the database, which leads to the threat of information disclosure at the data store; little or a weak deniable encrypted is used to protect the data, such that data can be proven to be an encryption or can be decrypted to a valid plaintext; and subjects with deniability are not able to edit data in the database to cover their tracks, and it can be either impossible to remove or alter the user's own data or impossible to remove or alter someone else's data concerning the user himself.

2.6.11 Non-repudiation of Process

Non-repudiation of process, as depicted in Figure 2.15 can be achieved in two ways: either the process loses its confidentiality and information disclosure attacks at the process are possible, or the process uses a secure log to create an overview of all actions, which can evidently be traced back to the user.

2.6.12 Detectability of Data Flow

The threat tree of *detectability of data flow*, as depicted in Figure 2.16 suggests five preconditions for the threat to occur. One condition is that the system lacks a

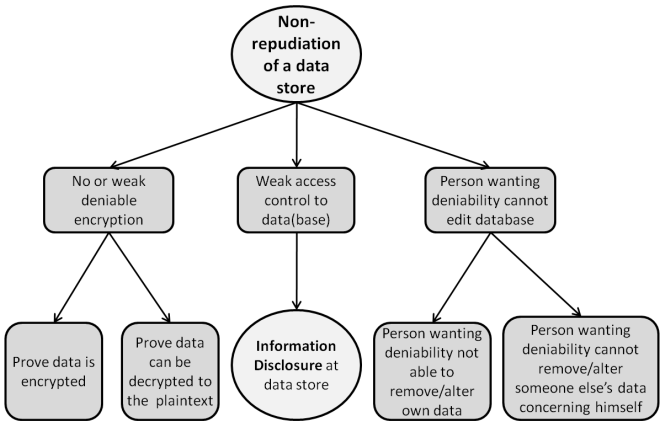


Figure 2.14. Threat tree for Non-repudiation of a data store

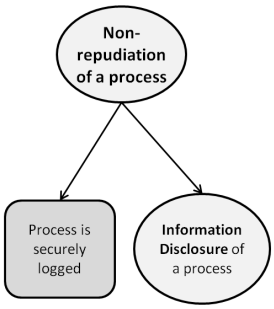


Figure 2.15. Threat tree for Non-repudiation of a process

covert channel. This can happen when the covert channel uses too much bandwidth from a legitimate channel, resulting in the detection of the covert communication. It can also be because the patterns or characteristics of the communications medium of the legitimate channel are controlled or examined by legitimate users, e.g. checking file opening and closing operations patterns or watching the timing of requests, such that covert communication is detected. The second condition is side channel analysis on timing information, power consumption, electromagnetic leaks, as an extra source of information which can be exploited to detect the communication. The third condition occurs when a weak information hiding techniques are used, which makes a number of steganalysis attacks possible. Another condition is when there is no or insufficient dummy traffic sent at some lower layer of communication network, such that messages fail to appear random

for all parties except the sender and the recipient(s). Last but not least, the threat can occur because of a weak spread spectrum communication, resulting in deficiencies in the establishment of secure communications, resistance to natural interference and jamming, and detection prevention.

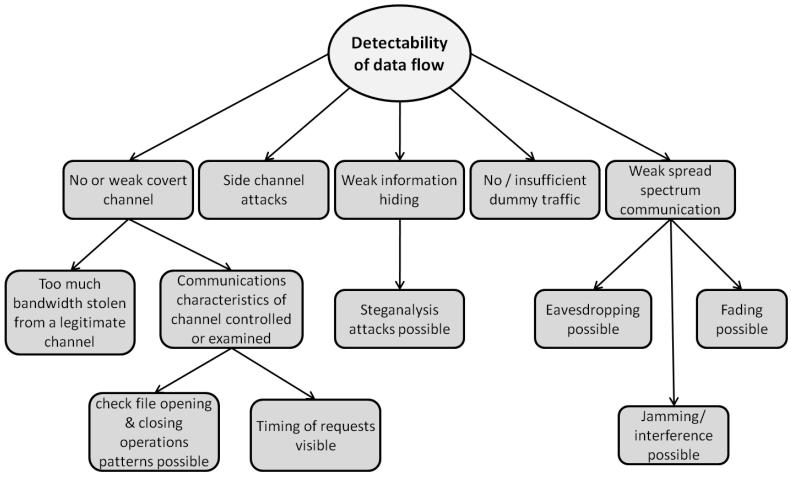


Figure 2.16. Threat tree for detectability of a data flow

2.6.13 Detectability of Data Store

As shown in Figure 2.17, *detectability threats in a data store* can occur if there is insufficient access control, which leads to the information disclosure threats for security, or if insufficient information hiding techniques are applied, such as information from a data store is revealed due to weak steganography algorithms employed.

2.6.14 Detectability of Process

Similar to the previously described threats related to a process, the *detectability of process* threat, depicted in Figure 2.18, also refers to the threat of information disclosure of process.

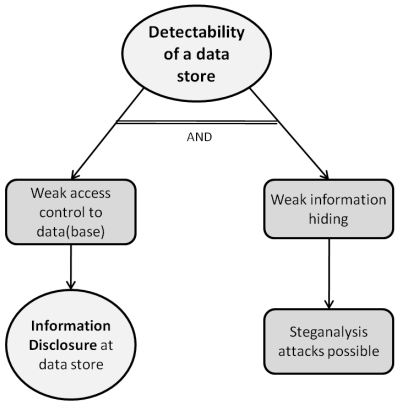


Figure 2.17. Threat tree for detectability of a data store

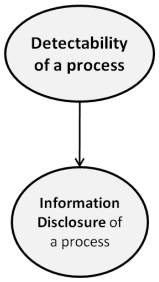


Figure 2.18. Threat tree for detectability of a process

2.6.15 Information Disclosure of Data Flow, Data Store, and Process

The threat tree concerning *information disclosure of data flow, data store, and process*, depicted in Figure 2.19, refers to the security threat tree of information disclosure. This illustrates the fact that privacy properties are part of security properties, and privacy may depend on security. For more information about the information disclosure threats we refer to SDL [163].

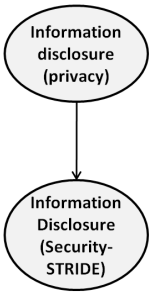


Figure 2.19. Threat tree for Information Disclosure

2.6.16 Content Unawareness of Entity

Content unawareness of an entity can occur in two situations: either the data subject provides more personal identifiable information than required, which has a negative influence on all the hard privacy objectives; or the data subject does not keep information updated or does not remove the outdated information, and it can lead to wrong actions when decisions are based on this information.

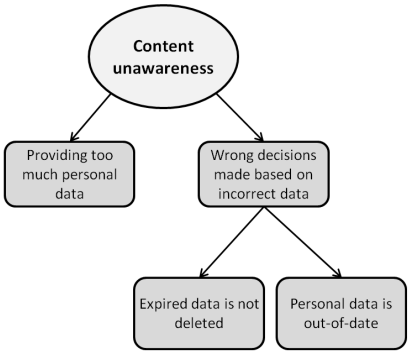


Figure 2.20. Threat tree for Content Unawareness

2.6.17 Consent and Policy Noncompliance of The System (Data Flow, Process and Data Store)

The user's privacy can be violated when internal system rules do not correspond to privacy policies provided to the user. This can occur when an attacker tampers

with the internal policies, which is actually a security threat; or when the policy rules are incorrectly managed or updated (according to the user's requests) by the system administrator.

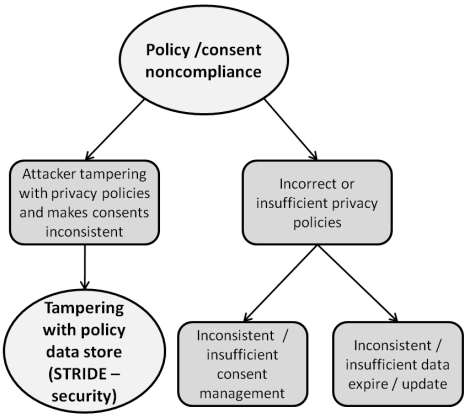


Figure 2.21. Threat tree for policy and consent noncompliance

2.7 From DFD and Privacy Threat Trees to Misuse Cases

2.7.1 Risk Assessment

Similarly to STRIDE, LINDDUN can suggest a (large) number of documented threats. Before the process moves forward, the identified threats must be prioritized. Only the important ones should be considered for inclusion in the requirements specification and, consequently, in the design of the solution. Risk assessment techniques provide support for this stage. In general, risk is calculated as a function of the likelihood of the attack scenario depicted in the MUC (misuse case) and its impact. The risk value is used to sort the MUCs: the higher the risk, the more important the MUC is.

The LINDDUN framework (similarly to STRIDE) is independent from the specific risk assessment technique that is used. The analyst is free to pick the technique of choice, for instance the OWASP's (Open Web Application Security Project) Risk Rating Methodology [228], Microsoft's DREAD [191], NIST's (National Institute of Standards and Technology) Special Publication 800-30 [221], or SEI's (Software Engineering Institute) OCTAVE [169]. These techniques leverage the information

contained in the MUC, as the involved assets (for the impact), and the attacker profile as well as the basic/alternative flows (for the likelihood). Many of the above-mentioned techniques include privacy considerations when assessing the impact of a threat. However, as a research challenge, a privacy-specific risk assessment technique is worthwhile to be investigated, as the on-field experience reveals any inadequacy of state-of-the-art techniques. This goes beyond the scope of this work.

2.7.2 Documenting Threats Scenarios in Misuse Cases

Threat tree patterns are used to detail the generic LINDDUN threat categories into specific threat instances that can occur in a system. Furthermore, some threat instances could have been discarded during the risk-analysis step. The result of the above process should be a collection of threat scenarios that need to be documented. To this aim, misuse cases can be used. In particular, a misuse case can be considered as a use case from the misactor's point of view. A misactor is someone who intentionally or unintentionally initiates the misuse case. Alexander [46] provides some example misuse cases, together with the corresponding (positive) use cases. We chose misuse cases because they represent a well established technique to elicit requirements and support tools exist as well.

The structure of a misuse case, which is based on the template provided by Sindre and Opdahl [51] is described below:

Misuse Case Structure

Summary: provides a brief description of the threat.
Assets, stakeholders and threats: describes the assets being threatened, their importance to the different stakeholders, and what is the potential damage if the misuse case succeeds.
Primary misactor: describes the type of misactor performing the misuse-case. Possible types are insiders, people with a certain technical skill, and so on. Also, some misuse case could occur accidentally whereas other are most likely to be performed intentionally.
Basic Flow: discusses the normal flow of actions, resulting in a successful attack for the misactor.
Alternative Flows: describes the other ways the misuse can occur.
Trigger: describes how and when the misuse case is initiated.
Preconditions: precondition that the system must meet for the attack to be feasible.

The preconditions refer to the leaf nodes of the threat tree patterns and the basic (and alternative) flow describes how a miuser could exploit these weaknesses of the system in order to mount an attack.

Running example: Social Network 2.0

In our running example, we assume that communication and processes within the social network service provider are trustworthy (see the trust boundary in the DFD depicted in Figure 2.1). However, we want to protect the data store against information disclosure. The data controllers could be users, social network providers, and application providers.

To illustrate how to create a misuse case based on the threat tree patterns, consider the threat tree of linkability at the data store (see Figure 2.7). The tree illustrates that in order to be susceptible to this threat, neither the data store is sufficiently protected against information disclosure nor sufficient data anonymization techniques are employed. These are the preconditions of the misuse case. To create the attack scenarios, it is clear that the attacker first needs to have access to the data store, and secondly, either the user (as the data subject) can be re-identified (as the basic flow) or the pseudonyms can be linkable (as the alternative flow). The aforementioned misuse case is presented in this section. The additional nine misuse cases applicable to the social network example are described in Appendix A.

MUC 1 – Linkability of Social Network Database (Data Store)

Summary: Data entries can be linked to the same person (without necessarily revealing the person's identity)
Assets, stakeholders and threats: Personal Identifiable Information (PII) of the user.

- The user:
    1. Data entries can be linked to each other which might reveal the person's identity
    2. The misactor can build a profile of a user's online activities (interests, actives time, comments, updates, etc.)

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor gains access to the database

2. The misactor can link the data entries together and possibly re-identify the data subject from the data content

Alternative Flow:

1. The misactor gains access to the database
2. Each data entry is linked to a pseudonym
3. The misactor can link the different pseudonyms together (linkability of entity)
4. Based on the pseudonyms, the misactor can link the different data entries

Trigger: by misactor, can always happen.
Preconditions:

1. no or insufficient protection of the data store
2. no or insufficient data anonymization techniques or strong data mining applied

Note that formulating soft privacy threats is less straight-forward and requires some out-of-the-box thinking for suitable (non-)technical solutions. We refer the reader to misuse cases 9 and 10 in the appendix as an example of the latter case.

2.8 From Threat Analysis to Privacy Enhancing Solutions

This section explains the elicitation of privacy requirements from threat analysis and the selection of mitigation strategies and techniques based on privacy objectives.

2.8.1 Eliciting Privacy Requirements: From Privacy Threat Analysis to Mitigation Strategy

Misuse cases describe the relevant (risk-wise) threat scenarios for the system. The preconditions are based on the threat tree patterns and the basic and alternative

Table 2.6. Privacy objectives based on LINDDUN threat types (E-Entity, DF-Data Flow, DS-Data Store, P-Process)

LINDDUN threats	Elementary privacy objectives
Linkability of (E, E)	Unlinkability of (E, E)
Linkability of (DF, DF)	Unlinkability of (DF, DF)
Linkability of (DS, DS)	Unlinkability of (DS, DS)
Linkability of (P, P)	Unlinkability of (P, P)
Identifiability of (E, E)	Anonymity / pseudonymity of (E, E)
Identifiability of (E, DF)	Anonymity / pseudonymity of (E, DF)
Identifiability of (E, DS)	Anonymity / pseudonymity of (E, DS)
Identifiability of (E, P)	Anonymity / pseudonymity of (E, P)
Non-repudiation of (E, DF)	Plausible deniability of (E, DF)
Non-repudiation of (E, DS)	Plausible deniability of (E, DS)
Non-repudiation of (E, P)	Plausible deniability of (E, P)
Detectability of DF	Undetectability of DF
Detectability of DS	Undetectability of DS
Detectability of P	Undetectability of P
Information Disclosure of DF	Confidentiality of DF
Information Disclosure of DS	Confidentiality of DS
Information Disclosure of P	Confidentiality of P
Content Unawareness of E	Content awareness of E
Policy and consent Noncompliance of the system	Policy and consent compliance of the system

flows are inspired by the system's use cases.

As a next step, the system's (positive) requirements can be extracted from the misuse cases. To this aim, the specification of the privacy requirements is facilitated by Table 2.6, which maps the types of threats scenarios to types of privacy requirements. Note that the table is a refinement of the more generic objectives in Table 2.2.

2.8.2 From Privacy Requirements to Privacy Enhancing Solutions

Similarly to security, privacy requirements can be satisfied via a range of solution strategies:

1. *Warn the user* could be a valid strategy for lower risk (but still relevant) threats. However precautions have to be taken so that users, especially nontechnical ones, do not make poor trust decisions.
2. *Removing or turning off the feature* is the only way to reduce the risk to zero. When threat models indicate that the risk is too great or the mitigation techniques are untenable, it is best not to build the feature in the first place, in order to gain a balance between user features and potential privacy risks.
3. *Countering threats with either preventive or reactive privacy enhancing technology* is the most commonly used strategy to solve specific issues.

This section mainly focuses on the last strategy. When countering threats with technology is chosen as the mitigation strategy, system designers have to identify the sound and appropriate privacy enhancing technology (PET). We summarize the state-of-art PETs in Table 2.7 and map these techniques to each of the corresponding privacy requirements of Table 2.6. As a result, improved guidance is provided to the system designers over the solution selection process.

Note that the PETs categorization is inspired by the taxonomies proposed in [294, 173]. Further, Table 2.7 introduces some key primitives of hard privacy technologies and the state-of-art of soft privacy technologies. New privacy enhancing solutions keep emerging; therefore a complete list of PETs and best practices for choosing the appropriate mitigation is beyond the scope of this chapter. The latest development of privacy enhancing technologies can be found at [8].

Table 2.7. Mapping privacy objectives with privacy enhancing techniques (U-Unlinkability, A-Anonymity/Pseudonymity, P-Plausible deniability, D-Undetectability/unobservability, C-Confidentiality, W-Content awareness, O-Consent/policy compliance of system)

	Mitigation techniques: PETs	U	A	P	D	C	W	O
Anonymity system	Mix-networks (1981) [93], DC-networks (1985) [94, 95], ISDN-mixes [238], Onion Routing (1996) [154], Crowds (1998) [256], Single proxy (90s) (Penet pseudonymous remailer (1993-1996), Anonymizer, SafeWeb), anonymous Remailer (Cipherpunk Type 0, Type 1 [58], Mixmaster Type 2 (1994) [4], Mixminion Type 3 (2003) [5]), and Low-latency communication (Freedom Network (1999-2001) [59], Java Anon Proxy (JAP) (2000) [67], Tor (2004) [132])	×	×			×		
	DC-net & MIX-net + dummy traffic, ISDN-mixes [238]	×	×		×	×		
	Broadcast systems [239, 286] + dummy traffic	×	×		×			
Privacy preserving authentication	Private authentication [40, 45]	×	×					
	Anonymous credentials (single show [76], multi-show [86])	×	×					
	Deniable authentication [214]	×	×	×				
	Off-the-record messaging [75]	×	×	×		×		
Privacy preserving cryptographic protocols	Multi-party computation (Secure function evaluation) [295, 215]	×				×		

	Anonymous buyer-seller watermarking protocol [115, 258]	×	×		×	
Information retrieval	Private information retrieval [98] + dummy traffic	×	×	×		
	Oblivious transfer [254, 82])	×	×		×	
	Privacy preserving data mining [285, 243]	×	×		×	
	Searchable encryption [41], Private search [227]		×		×	
Data anonymization	K-anonymity model [276, 275], l-Diversity [196]	×	×			
Information hiding	Steganography [50]	×	×	×		
	Covert communication [212]	×	×	×		
Pseudonymity systems	Privacy enhancing identity management system [158]	×	×			
	User-controlled identity management system [99]	×	×			
	Privacy preserving biometrics [269]	×	×		×	
Encryption and signature techniques	Symmetric key & public key encryption [204]				×	
	Deniable encryption			×	×	
	Homomorphic encryption [230, 108]				×	
	Verifiable encryption [84]				×	
	Group Signature [66, 65]	×	×			
Access control techniques	Context-based access control [151]				×	
	Privacy-aware access control [91, 54]				×	
Policy and feedback tools	Policy communication (P3P [229])					×

Policy enforcement (XACML [222], EPAL [165])			×
Feedback tools for user privacy awareness [188, 231, 192]		×	
Data removal tools (spyware removal, browser cleaning tools, activity traces eraser, harddisk data eraser)			×

In summary, privacy protection solutions boil down to either technical or legal enforcement. In general, privacy technology enables functionality while offering the highest protection for privacy. Further, *Hard Privacy Technology* provides cryptographically strong protections for privacy, assumes no unnecessary leakage of information, and replies on massive distribution of trust excluding potential adversary and privacy violators. *Soft Privacy Technology* (e.g. privacy policy and feedback tools in Table 2.7) offers protections against mass surveillance and violations, assumes data subjects sharing of personal data is necessary, and employs a weaker adversary model.

Running example: Social Network 2.0

Table 2.8 summarizes the selection of PETs based on the privacy requirements elicited in our running example. It is possible that a more business-oriented example would suggest different mitigation strategies. Nevertheless, we hope the example depicted in this section illustrates how the proposed framework can be applied in real life applications.

In an attempt to make the running example more accessible to the reader, the system model, the misuse cases, and the mitigation techniques of the Social Network 2.0 are largely simplified due to the assumption that the social network providers are semi-trustworthy (i.e., the adversary model consists of external parties, data holder, honest insiders who make errors, and corrupt insiders). If different assumptions would hold, different misuse cases should be identified with a distinct mitigation approach. For instance, if we apply a smaller trust boundary and assume that the social network provider is totally untrustworthy, then extra privacy requirements and a stronger threat model would be considered. One possible misuse case would be that the malicious social network provider, as an attacker, takes advantage of profiling user's personal data for its own benefits. In that scenario, one solution could be building a security architecture out of smart clients and an untrusted central server to removes the need for faith in network operators and gives users control of their privacy [49]. Another solution could be using encryption to enforce access control for users' personal information based on their privacy preferences [63, 249].

Table 2.8. Social Network 2.0 example: from misuse cases to privacy requirements and suggested mitigation strategies and techniques

No.	Misuse cases	Privacy requirements	Suggested mitigation strategies and techniques
1	Linkability of social network data store	Unlinkability of data entries within the social network database	Apply data anonymization techniques, such as k-anonymity [275].
		Protection of data store	Enforce data protection by means of relationship-based access control [91]
2	Linkability of data flow of the user data stream (user-portal)	Unlinkability of messages of user-portal communication; channel confidentiality	Deploy anonymity system, such as TOR [132].
3	Linkability of entities the social network users	Unlinkability of different pseudonyms (user IDs) of social network users; channel confidentiality.	1) Technical enforcement: deploy anonymity system, such as TOR [132], for communication between user and social network web portal; 2) User privacy awareness: inform users that revealing too much information online can be privacy invasive.
4	Identifiability at the social network data store	Anonymity of social network users such that the user will not be identified from social network database entries	Protection of the data store, by applying data anonymization techniques, such as k-anonymity [275].
		Protection of data store	Enforce data protection by means of relationship-based access control [91]

5	Identifiability at data flow of user data stream (user-portal)	Anonymity of social network users such that the user will not be identified from user-portal communication by content; channel confidentiality	Deploy anonymity system, such as TOR [132], for communication between user and social network web portal.
6	Identifiability of the social network users	Pseudonymize users IDs	1) Apply secure pseudonymization techniques to issue pseudonyms as user IDs; 2) User privacy awareness: inform users using real ID has a risk for privacy violation.
		Use identity management to ensure unlinkability is sufficiently preserved (as seen by an attacker) between the partial identities of an individual person required by the applications	Employ privacy preserving identity management, e.g. proposed in [158], together with user-controlled identity management system [99] to ensure user-controlled linkability of personal data. System supports the user in making an informed choice of pseudonyms, representing his or her partial identities. Make the flow of this user's identity attributes explicit to the user and gives its user a large degree of control.
		Confidentiality of data flow in user-portal communication	Deploy anonymity system such as TOR [132].
7	Information disclosure at the social network data store	Release of the social network data store should be controlled according to user's privacy preference	Apply access control at the social network databases, e.g. privacy aware collaborative access control based on relationships [91]

8	Information disclosure of communication between the user and the social network	Confidentiality of communication between the user and the social network should be ensured	Employ a secure communication channel and deploy anonymity system such as TOR [132].
9	Content unawareness of user	Users need to be aware that they only need to provide minimal set of required personal data (the data minimization principle)	Use feedback tools to raise user's privacy awareness.
10	Policy and consent noncompliance of the whole social network system	Design system in compliance with legal guidelines for privacy and data protection	1) Hire employee who is responsible for making the policies compliant OR hire external company for compliancy auditing 2) Ensure training obligations for employees.
		Ensure user aware that in case of violation, user is legitimated to take legal actions	E.g., user can sue the social network provider whenever user's personal data is not processed according to what is consented.
		Employee contracts clearly specify do's and don'ts according to legal guidance	1) Ensure training obligations for employees; 2) Employees who disclose users information will be penalized (get fired, pay fine, etc.).

Another research discussion is concerning practicality to build user privacy feedback tools. In short, from a technical point of view, feedback could be realized by means of data mining techniques (e.g., k-anonymity model) to countermeasure user identification and data profiling attacks. It compares data user sends to the social network with a whole set of data composed of data from all networks users, and checks the "uniqueness" of personal identifiable information (PII) of the user. With a unique PII, a user has a higher probability to be identified. Then it warns users each time their activities provoke privacy risks, e.g. shows a risk level of identifiability by posting a message "you are about to leave the anonymity safe zone". There are some research incentives for feedback systems for social networks [188, 231, 192]. However, this concept implies a paradox that in order to ensure accurate feedback, the feedback tool itself should be a "perfect attacker" that knows all the data from all users. Due to the space and scope limit of this chapter, we cannot discuss this in detail. We encourage interested readers to formalize the feedback system model and investigate whether it is technically realistic to realize the feedback concept and beyond which threshold a feedback could be satisfactory. Intuitively speaking, the aforementioned feedback concept is not about technical problem purely but more an education problem to raise user's privacy awareness. The usability of such feedback tools is also an issue, such as how to design a user friendly interface and encourage users to use feedback remains a research challenge.

2.9 Discussion

Despite the fact that the dualism between hard and soft privacy has already been generically introduced in a few talks [109, 111], to our best knowledge, this work is the first effort that concretely distinguishes these two concepts and categorizes privacy properties (and threats) accordingly. In short, hard privacy properties include unlinkability, anonymity and pseudonymity, plausible deniability, undetectability and unobservability, and confidentiality. Soft privacy properties include content unawareness and policy and consent compliance. Similarly, leveraging the link between privacy enhancing technologies and privacy objectives, it is possible to make a distinction between hard and soft solutions. Hard privacy technologies are active in research but inadequate in deployment, due to cost and technical evolvement restrictions (such as cryptography). Soft privacy technologies have fewer research activities. With legal compliance as a strong driver, soft privacy solutions reply on stakeholder's liability and the tradeoffs between cost of deploying privacy solutions and potential costs in case of massive data breach. After all, building in privacy in the system might not be cheap, but just cheaper than building in no privacy.

We gained an insight into the proposed methodology that is worthy to be emphasized:

1. Some privacy threats, in contrast to security threats, affect DFD elements pair-wise (or sometimes group-wise). For instance, unlinkability implies the relation of two or more items of interest. As an example, a relation may refer to a subject (an entity in the DFD) and its attributes (in the data stores). Consequently, it is straightforward to see that linkability threats always affect a pair or a group of DFD elements. Similarly, plausible deniability refers to the pair-wise relation between a subject and the attribute that the subject wants to deny. We can draw a conclusion that privacy emphasizes relationships between instances of DFD elements (e.g. two communication instances of the same data flow cannot be linked) or relationships between a DFD element and an entity, while security focuses on each individual DFD component in a more local way.

2. The *process* element in the DFD is less important for privacy because privacy cares more about the relationships between entities and data. It is quite the opposite in the case of security. For instance, all STRIDE threat categories apply to the process element, while, in LINDDUN, all privacy attack paths involving the process element are related to a security threat (namely, information disclosure of process) and not to privacy threats per se. This observation leads to our next finding.

3. The authors have chosen the DFD notation to represent a system in order to keep compliance with the STRIDE approach. As STRIDE and LINDDUN are expected to be applied in a synergic way, this choice fosters the reuse of the DFD models (and the modeling knowledge) across the security and the privacy threat analysis.

   However, it should be noticed that DFD elements (entities, processes, data flows, and data stores) represent the technical assets that require protection from privacy-specific harm. That is, the DFDs expose the privacy-relevant information concerning the technical assets from the system perspective. Therefore, the privacy analyst should pay attention to the fact that the important privacy assets are properly modeled by the DFD. For instance, data flows should be specific to IOIs in the privacy case.

4. For some privacy properties, an extension of the DFD semantics might be useful. For instance, for the case of privacy-sensitive information that is inferred over time, the notion of knowledge is necessary. This could be annotated in the DFD processes via epistemic constructs and then leveraged during the analysis.

5. Concerning the privacy threat trees, one can see that many paths lead to security threats (e.g. information disclosure and tampering). Consequently,

privacy objectives heavily depend on security objectives (e.g. confidentiality and integrity of data flow, data store or process). We draw the general conclusion that it is interesting to analyze privacy threats together with security threats and the necessary process (and tool) support should be built to facilitate such activities. This synergy would bring an advantage in terms of time and cost for system designers to work with. Further, privacy and security objectives might conflict (e.g. non-repudiation and plausible deniability, as explained in the previous sections). It is thus useful to perform the threat analysis for privacy and security altogether and consider both types of requirements at the same time.

Finally, it is necessary to clarify a few definitions to avoid confusion. Confidentiality refers to hiding the data content; anonymity and pseudonymity refer to hiding the subject's identity or hiding the link between identity and action or a piece of information; unlinkability refers to hiding links between two or more objectives (actions, identities, and pieces of information); and undetectability and unobservability refers to hiding a subject's activity. In particular, unlinkability of entities means that it is impossible to relate different entities based on some common personal identifiable information (PII), while anonymity of entities means that the subject cannot be identified within an anonymity set. Anonymity at data flow typically means that, in an anonymous communication setting, the subject cannot be identified from the data flow's content or side channel information; while anonymity at data store refers to data anonymization, meaning that the subject cannot be identified from the content of the data store.

2.10 Conclusion

In this chapter, we have presented a comprehensive framework to model privacy threats in application systems, elicit privacy requirements, and instantiate privacy enhancing countermeasures. The primary contribution is the systematic methodology to model privacy specific threats. This is achieved by defining a list of privacy threat types and providing the necessary mappings to the elements in the system model. The second contribution is represented by the supporting body of knowledge, namely, an extensive catalogue of privacy specific threat tree patterns. In addition, this work provides the means to map the most commonly known privacy enhancing technologies (PETs) to the identified privacy threats and the elicited privacy requirements. The privacy threat tree patterns and categorization of suggested PETs are expected to be continuously updated and improved upon, since new threats keep emerging, just as privacy technologies keep developing.

Chapter 3

Anonymous Buyer-Seller Watermarking Protocols

3.1 Introduction

Today's rapid proliferation of computer networks and multimedia technology facilitates the efficient distribution of multimedia content. However, it also eases the reproduction and the distribution of illegal copies, and raised a number of security issues including copyright protection, traitor tracing, entity and data authentication. At the same time, more attention has been paid to privacy protection for users in emerging multimedia applications. Therefore, the development of techniques in order to meet these needs has become an important concern.

3.1.1 Previous Work

Digital watermarking and fingerprinting techniques have experienced a surge in research activities over the last decade, and a variety of elegant watermarking (fingerprinting) protocols have been proposed [241, 240, 83], allowing content providers to embed the provider's information in a distributed content to preserve the copyright, or a customer's information to identify copyright violators. Fingerprinting schemes have been proposed to identify different kinds of digital content, such as documents [77, 74], images or videos [287, 281, 193], or computer programs [157]. A first improvement of fingerprinting techniques was the design of collusion-resistant schemes [69, 74, 281], i.e., schemes that tolerate a collusion of buyers up to a certain size by preventing colluding buyers that compare their

different copies from creating a copy that cannot be traced back to one of the colluders.

Traditional watermarking based fingerprinting schemes assume that content providers are trustworthy such that they would never distribute content illegally and always perform the watermark embedding honestly. Unfortunately, in practice, such assumptions are not fully established. This problem was first identified by Qian and Nahrstedt as the *customer's rights problem* [253], where the watermark is generated and embedded solely by the content provider (or the seller). A customer (or the buyer) whose watermark has been found in unauthorized copies can claim that the pirated copy was created by the seller. This could be done for instance by a malicious seller who may be interested in framing the buyer. It could be also possible when the seller is not the original owner but a reselling agent who could potentially benefit from making unauthorized copies. Finally, even if the seller was not malicious, an unauthorized copy containing the buyer's fingerprint could have been originated from a security breach in the seller's system but not from the buyer [253].

The owner-customer watermarking protocol proposed by Qian and Nahrstedt [253] tries to solve this problem such that the customer provides the owner with an encrypted predetermined bit-string, and the owner embeds the encrypted value using an invisible watermarking technique. Upon receiving the watermarked content delivered from the owner, the customer is able to prove to a third party the legitimate ownership of the copy in the customer's possession, since only the buyer knows the decryption key. The drawback of this protocol is that it doesn't solve the problem of irrevocable binding the customer and the specify copy sold to him, and holding the customer responsible for any unauthorized copies of the same found in the market. This is due the problem of traditional symmetric fingerprinting schemes, where both buyer and seller know the copy that the buyer gets. In symmetric schemes, a malicious seller can release a pirated copy in order to frame an honest buyer, and a guilty buyer can repudiate the accusation of copyright infringements by invoking the possibility of being framed by the seller or caused by a security breach in the seller's system. As a consequence, the watermark tracing mechanism is discredited.

It is against this background that asymmetric schemes [241, 242, 68] were introduced, where only the buyer obtains the exact watermarked content, and hence the buyer cannot claim that a pirated copy was originated from the seller. In the asymmetric fingerprinting protocol proposed by Pfitzmann and Schunter [241], the buyer chooses a secret and sends a commitment to the secret to the seller. Then buyer and seller execute a protocol at the end of which the buyer obtains a watermarked content with the buyer's secret, while the seller does not get any information. Therefore, when the seller is able to provide the secret chosen by the buyer, it must be the case that he found a pirated copy, and thus the buyer is found guilty.

In the aforementioned symmetric and asymmetric schemes the buyer needs to be authenticated by the seller at each purchase. To protect buyers' privacy, Pfitzmann and Waidner [242] introduced anonymous asymmetric schemes, where buyers remain anonymous as long as they do not release pirated copies. Buyers are required to register at a registration entity prior to any purchase and, if the seller finds a pirated copy, he can query this registration entity to revoke buyers' anonymity. First anonymous asymmetric schemes [42] require interaction with the buyer in case of dispute to find out whether the buyer was guilty or not guilty. Pfitzmann and Sadeghi [240] and Camenisch [83] proposed schemes that allow direct non-repudiation, where the seller, upon finding a pirated copy, possesses enough information to convince a third party of the buyer's culpability.

3.1.2 Basic Concept

Incorporating cryptography with digital watermarking, a *buyer-seller watermarking (BSW) protocol* is in fact an asymmetric fingerprinting protocol where the fingerprint is embedded by means of watermarking in the encrypted domain. The basic idea is that each buyer obtains a slightly different copy of the digital content offered by the seller. Such a difference, the watermark (or fingerprint), does not harm the perceptual quality of the digital content and cannot be easily removed by the buyer. Thanks to the latter property, when a malicious buyer redistributes a pirated copy, the seller can associate the pirated copy to its buyer by its embedded watermark. On the other hand, a malicious seller cannot frame an honest buyer because the buyer's watermark and the delivered watermarked content are unknown to the seller. A complete and sound buyer-seller watermarking protocol is expected to solve the following BSW problems:

1. **The piracy tracing problem.** Once a pirated copy is found, the seller should be able to trace and identify the copyright violator.

2. **The customer's rights problem.** When a watermark is inserted solely by the seller, the seller may benefit from framing attacks to an innocent buyer. This also causes unsettled disputes. On the other hand, the buyer accused of distributing an unauthorized copy may claim that the copy originated from the seller or that there existed a security breach in the seller's system.

3. **The unbinding problem.** Upon discovering a pirated copy, the seller can fabricate piracy by transplanting the buyer's watermark into other digital content. Therefore, it is necessary to bind a chosen watermark with a specific transaction.

4. **The anonymity problem.** The identity of a buyer should remain unexposed during transactions unless he is proven to be guilty.

Table 3.1. Comparison of some existing buyer-seller watermarking protocols with our protocols. Problems solved by each protocol: 1. (Piracy tracing problem), 2. (Customer's rights problem), 3. (Unbinding problem), 4. (Conspiracy problem), 5. (Dispute resolution problem), and 6. (Anonymity/unlinkability problem)

Problems Solved	1.	2.	3.	4.	5.	6.
Memon and Wong's protocol [203]	√	√				
Ju et al.'s protocol [172]	√	√			√	
Choi et al.'s protocol [170]	√	√				
Goi et al.'s protocol [152]	√	√		√		
Lei et al.'s protocol [189]	√	√	√		√	
Zhang et al.'s protocol [299]	√	√	√	√		
Shao et al.'s protocol [268]	√	√	√		√	
Ibrahim et al.'s protocol [166]		√	√		√	
Our protocol	√	√	√	√	√	√

5. **The conspiracy problem.** Malicious parties may collude with each other and mount attacks to frame an innocent buyer or to confound the tracing by removing the watermark from the digital content.

6. **The dispute problem.** The arbitrator should be able to resolve disputes, without the buyer revealing her identity or private key.

3.1.3 Existing BSW Protocols

The literature is rich of relevant buyer-seller watermarking protocols. Since the introduction of the concept by Memon and Wong [203], a number of buyer-seller watermarking protocols have been proposed. This section summarizes some previously proposed protocols by Memon and Wong [203], Ju et al. [172], Choi et al. [170], Goi et al. [152], Lei et al. [189], Zhang et al. [299], Shao et al. [268], and Ibrahim et al. [166]. The comparison of the related work with our proposed protocols is depicted in Table 3.1.

- **The piracy tracing problem.** All of these protocols are able to resolve the piracy tracing problem, and provide a mechanism for the seller to trace and recover the identity of a guilty buyer.

- **The customer's rights problem.** All these protocols can solve the customer's rights problem, since the protocols are designed to be asymmetric, i.e., the seller doesn't know the exact value of the buyer's watermark, neither does she know the final watermarked digital content that the buyer gets.

Therefore, the accused buyer for an illegal replication or distribution cannot claim that the copy is originated from the seller or a security breach in the seller's system.

- **The unbinding problem.** Lei et al. [189] addressed *the unbinding problem* in [203, 172, 170, 152] and provided a mechanism to bind a specific transaction of a digital content to a specific buyer, such that a malicious seller cannot transplant the watermark embedded in a digital content to another higher-priced content. The similar design principle is applied by Zhang et al. [299] and Shao et al. [268].

- **The conspiracy problem.** Choi et al. [170] pointed out the *conspiracy problem* in [203, 172], where a malicious seller can collude with an untrustworthy third party to fabricate piracy to frame an innocent buyer. Goi et al. [152] found the conspiracy problem couldn't be solved through commutative cryptosystems of [170], further pointed out that schemes of [203, 172, 170] are vulnerable against *conspiracy attacks*, and showed that the protocol's security shouldn't rely on any third party. Zhang et al. [299] applied the principle of [152] and ensured that the buyer's watermark is generated by the buyer, instead of a Watermark Certification Authority (WCA). According to our analysis, we found that the protocols by Lei et al. [189], Shao et al. [268], and Ibrahim et al. [166] cannot resist the conspiracy attack, where a malicious seller can collude with a third party, such that the seller can discover the buyer's watermark.

- **The anonymity problem.** The protocol by Memon and Wong [203] requires the seller to know the buyer's identity to carry out a transaction. Schemes of [172, 170] improved [203] by applying an anonymous key pair in each transaction. However, both protocols require the WCA to know the buyer's identity, which means that the buyer's anonymity is not preserved against conspiracy attacks. In [152], the buyer is required to request a signature from the certification authority (CA) of the public key infrastructure (PKI) to generate a watermark. However, [152] cannot solve the anonymity problem efficiently, since before each transaction, the buyer has to contact the CA for a new signature. The schemes of [189, 299, 268] employ anonymous certificates, i.e., digital certificates without real identities of applicants. Unfortunately, transaction unlinkability is not provided: during all transactions, the anonymous certificate remains the same, unless the buyer contacts the CA before each transaction for a new certificate, which is impractical for real life applications.

- **The dispute problem.** Zhang et al. [299] presented a scheme, derived from [189], in which no trusted third party (TTP) is required in the watermark generation phase and the conspiracy problem is solved. Unfortunately, we found the existence of *dispute resolution problem* in [299]: in order to

resolve disputes the buyer is required to cooperate and reveal his secret key or his secret watermark to the judge or to the *CA*. This is unrealistic in real-life applications. Similarly, schemes of [203, 170, 152] all require the accused but possibly innocent buyer to reveal his identity or private key. Moreover, these protocols don't operate properly if the underlying cryptosystem is probabilistic, since the data encrypted by the judge or the *CA* are not equal to the data provided by the seller. In [172], the buyer creates a key escrow cipher to escrow his anonymous private key at the judge. The problem of this scheme is that the buyer's secrecy could not be guaranteed against conspiracy attacks if the judge was malicious. In [189], the judge requests the buyer's watermark from the *WCA*, and hence the protocol's security depends on the trustworthiness of the *WCA*.

3.1.4 Summary of Contributions

The contributions of this chapter are summarized as follows:

Analysis and attacks of two BSW protocols

We present the attacks on two recently proposed BSW protocols by Lei et al. [189] and Ibrahim et al. [166], and prove that these protocols are not able to provide security for both the buyer and the seller simultaneously as they have claimed. Additionally, we show that neither of the protocols works properly with probabilistic homomorphic cryptosystems, which is an essential mechanism to provide security for buyer and seller, and to facilitate watermark embedding in the encrypted domain. Further, we point out that the protocol of Ibrahim et al. [166] is not able to provide buyer's anonymity and transactions unlinkability.

Security Definition of BSW Protocols

We provide a security definition for proposed BSW Protocols. Our definition is generic, in the sense that it captures the security properties required for any copyright protection protocol that provides buyers with revocable anonymity.

Type I BSW protocol

From the analysis of early BSW protocols in Section 3.1.3, we show that previous study mainly focuses on the copyright protection issues. However, the existing solutions to the anonymity protection or dispute resolution problems are either impractical or incomplete. We propose an anonymous buyer-seller watermarking

protocol that fulfills the design requirements, and we will name it the Type I BSW protocol. Different from the predecessors, the Type I BSW protocol makes improvements on the following aspects:

1. The watermark generation and embedding phase of the Type I BSW protocol is based on the proposal from Memon and Wong [203]. However, the proposed protocol extends and improves its predecessors, by incorporating homomorphic encryption schemes, group signature schemes, and anonymous communication channels, to ensure revokable anonymity for buyer and security for both buyer and seller. Our improvement on the protocol's security properties ensures that the BSW problems outlined in Section 3.1.2 are fulfilled. Homomorphic encryptions facilitate operations such as watermark embedding in the encrypted domain. Group signatures introduces piracy traceability, buyer's anonymity, and transactions unlinkability. Besides, anonymous communication channels enable both anonymous outgoing connections and anonymous hidden services. We assume that a public key infrastructure (PKI) is available, such that each party has a public and private key pair certified by a trustworthy registration entity such as the CA.

2. The Type I BSW protocol supports multiple transactions. The protocol consists of three phases, such as registration, purchase, and arbitration, which are presented by three subprotocols respectively, namely the registration protocol, the watermarking generation and embedding protocol, and the identification and arbitration protocol. Prior to the purchase phase, a buyer first joins a group by registering the buyer's identity and PKI certificates. In return, the group manager or a Trusted Registration Authority issues the buyer a private group signature key. During the purchase phase, the buyer may execute the watermark generation and embedding protocol multiple times with a number of sellers and obtain the desired digital content anonymously.

3. The Type I BSW protocol doesn't require buyers to participate in the dispute resolution phase. Once a pirated copy is found, the corresponding seller will extract the watermark from the copy and check if it would be related to any buyer from a particular transaction. If a dispute resolution is necessary, the seller will go to a trustworthy a legal institution such as a judge from a civil court for arbitration. Based on the recorded transaction information provided by the seller, the judge is able to arbitrate whether the suspected buyer has indeed created the unauthorized copy or not. If that would be the case, the judge should send a court order to the Trusted Registration Authority (or the group manager) to recover the buyer's identity. At the end of arbitration, the judge is able to inform the seller whether the buyer is guilty or not, and the identity of the guilty buyer.

Type II BSW protocol

The Type II BSW protocol improves the Type I BSW protocol in the watermark generation and embedding phase. Besides the improvements introduced, the Type II BSW protocol also has the following advantages:

1. In the purchase phase of this protocol, both the buyer and the seller generate their secret watermarks. With the buyer's encrypted watermark and the seller's watermark, the seller is able to compute a composite watermark. Next, the seller performs a double watermark embedding: first the seller embeds a unique watermark to the original content in the plaintext domain, and then embeds the composite watermark in the encrypted domain. In this scheme, none of the parties know the exact watermark embedded in the original content, and only the buyer knows the final watermarked content that the buyer obtains.
2. One of the main differences between the Type II BSW protocol and the Type I BSW protocol is that the underlying watermarking scheme is not limited to permutation tolerant or linear watermarks, such that given the watermarked content Y_1 and Y_2, $\phi(Y)$ (ϕ is a permutation function) or $aY_1 + bY_2$ ($a, b \in \mathcal{R}$) is another valid watermarked content. In our protocol, we need a functionality such that, given a vector of encrypted watermark bits $\mathcal{E}[w_i]$ and a content X, it is able to produce the encrypted and watermarked content $\mathcal{E}[Y]$. Every watermarking scheme that is invisible and robust to counter post image processing or malicious attacks that are possibly encountered later, and supports the above functionality can be used with our scheme.

Type III BSW protocol

The Type III BSW protocol further improves the Type I and Type II BSW protocols in a number of aspects:

1. The Type III BSW protocol employs blind and readable watermarking schemes, homomorphic encryptions, group signature schemes and several zero-knowledge proofs of knowledge as main cryptographic building blocks. We prove the security of the protocol when instantiated with any secure watermarking scheme and with any secure building blocks.
2. The Type III BSW protocol doesn't require the seller to embed the watermarks twice in the purchase phase as most of the predecessors do. Double watermark insertions have the drawback of causing a degradation of the final quality of the distributed content, thus end up reducing their commercial value. When applied independently, the second watermark could

confuse or discredit the authority of the first watermark, thus acting as an actual "ambiguity attack" [145, 105]. We avoid it by designing a composite watermark, which is composed of the buyer's secret watermark, the seller's secret watermark, and a transaction index.

3. The Type III BSW protocol avoids the need for the seller to send the unauthorized content and the secret watermarking key to the judge and, as a consequence, reduces the communication bandwidth. In the arbitration phase, instead of the judge obtaining the unauthorized copy and detecting the watermark from the unauthorized content, the seller sends the watermark that she extracted from the pirated content to the Judge. In particular, the buyer's security (non-frameability) is based, among other properties, on the IND-CPA security of the homomorphic encryption scheme used to encrypt the buyer's secret watermark and the non-frameability property of the underlying group signature scheme. It means that the seller could only frame an honest buyer if the seller successful guessed the buyer's secret watermark. In order to do so, the seller needs to know the buyer's secret watermark. In this regard, it doesn't matter for the seller to send the pirated copy or to send the extracted watermark to the judge, since in either case, the seller cannot frame an honest buyer. In practice, this improvement reduces the communication bandwidth, e.g. sending a 128-bit watermark instead of a complete 2-Mbit image.

4. Another contribution of our work is a formal security analysis of BSW protocols. None of the previously proposed BSW protocols provides a formal security definition, and proves that the proposed protocol satisfies the required security properties. We employ the ideal-world/real-world paradigm [90] to define security of anonymous BSW protocols. Additionally, we define security for blind watermarking schemes and prove that the proposed Type III BSW protocol fulfills our security definition.

3.1.5 Details on Publications

The aforementioned contributions of this chapter have been published in [125] for the Type I BSW protocols, [123, 124, 126] for the Type II BSW protocols, [129, 115, 116] the Type II BSW protocols, where security proof and formal definition of BSW protocols is in [258]. Efficient implementation results of BSW protocols are briefly presented in this thesis. Please refer to [129, 115, 116, 117] for more details.

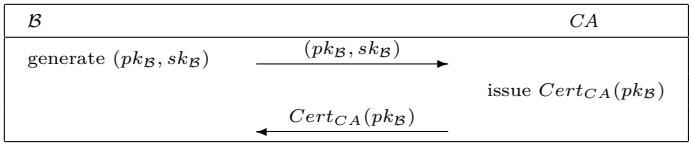


Figure 3.1. The registration phase of Lei et al.'s protocol

3.1.6 Chapter Outline

The rest of the chapter is organized as follows. First, two BSW protocols proposed by Lei et al. [189] and by Ibrahim et al. [166] are analyzed in Section 3.2. Section 3.3 briefly reviewed the definition and the properties of the cryptographic building blocks to be employed in the BSW protocols described in this chapter. Section 3.4 provides the generic security definitions for blind watermarking schemes and for BSW protocols. Our proposed Type I BSW protocol, Type II BSW protocol, and Type III BSW protocol are described in Section 3.5, Section 3.6 and Section 3.7, respectively. In Appendix B, we analyze the security of the Type III BSW protocol; Appendix C discusses the implementation of the Type III BSW protocol. We conclude in Section 3.8.

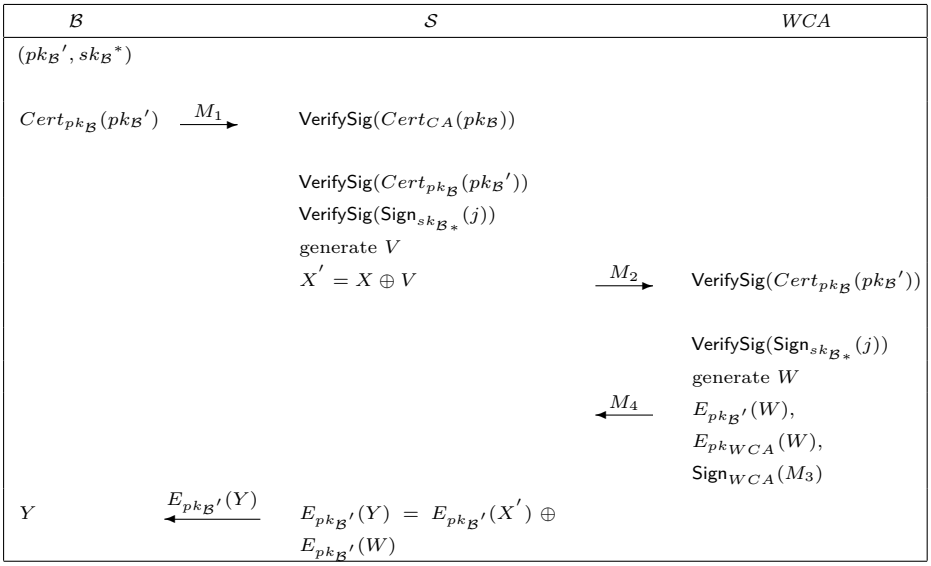
3.2 Attacks to existing protocols

3.2.1 Attacks on the Protocol of Lei et al.

In the protocol proposed by Lei et al. [189], the players are the seller $\mathcal{S}$, the buyer $\mathcal{B}$, the certificate authority CA, the watermark certificate authority WCA, and the judge $\mathcal{J}$. The protocol comprises three phases, namely the registration protocol, the watermark generation and insertion protocol, and the identification and arbitration protocol. The overview of the protocol is in Figure 3.1, Figure 3.2, and Figure 3.3.

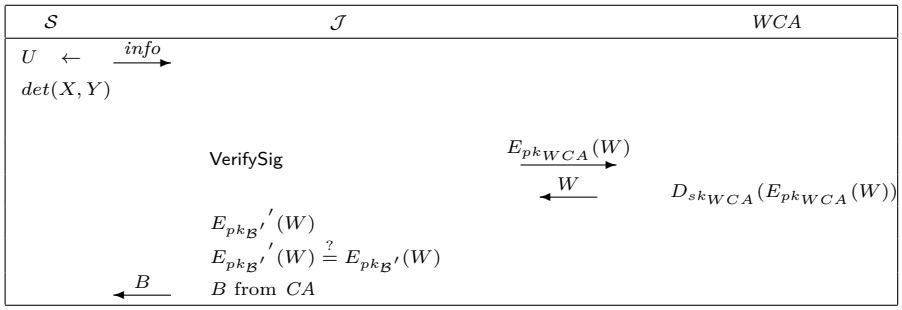
Attack Buyer's Security

In the protocol, the seller $\mathcal{S}$ generates her watermark V and embeds V to the original content X, as $X^{'} = X \oplus V$. The WCA generates $\mathcal{B}$'s watermark W, and sends $\mathcal{S}$ two encrypted values of W with $\mathcal{B}$'s encryption key $pk_{\mathcal{B}}{}'$ and WCA's encryption key, respectively. $\mathcal{S}$ embeds the encrypted watermarked, $E_{pk_{\mathcal{B}}{}'}(Y) = E_{pk_{\mathcal{B}}{}'}(X^{'}) \oplus E_{pk_{\mathcal{B}}{}'}(W)$.



Note: $M_1 = \{Cert_{CA}(pk_{\mathcal{B}}), Cert_{pk_{\mathcal{B}}}(pk_{\mathcal{B}}{}'), j, \mathsf{Sign}_{sk_{\mathcal{B}*}}(j)\}$
$M_2 = \{Cert_{pk_{\mathcal{B}}}(pk_{\mathcal{B}}{}', j, \mathsf{Sign}_{sk_{\mathcal{B}*}}(j), X'\}$
$M_3 = \{E_{pk_{\mathcal{B}}{}'}(W), pk_{\mathcal{B}}{}', \mathsf{Sign}_{sk_{\mathcal{B}*}}(j)\}$
$M_4 = \{E_{pk_{\mathcal{B}}{}'}(W), E_{pk_{WCA}}(W), \mathsf{Sign}_{WCA}(M_3)\}$

Figure 3.2. The watermark generation and insertion phase of Lei et al.'s protocol



Note: $info = \{Y, X' = X \oplus V, M_1, M_4\}$

Figure 3.3. The identification and arbitration phase of Lei et al.'s protocol

If the *WCA* would be untrustworthy, a malicious $\mathcal{S}$ could collude with the untrustworthy *WCA*, $\mathcal{S}$ sends $E_{pk_{\mathcal{B}'}}(W)$ back to the *WCA*. *WCA* recovers W via decryption, and sends W to $\mathcal{S}$. After $\mathcal{S}$ obtains W, she knows all the necessary information X, V, W to reproduce the watermarked content Y for $\mathcal{B}$. Once $\mathcal{S}$ gets $\mathcal{B}$'s watermark, any important features of the protocol would end up getting compromised. First, the piracy traceability won't be achieved, since both the buyer and the seller might be the traitor. Second, the non-framing property fails, though the unbinding problem is solved in the protocol. $\mathcal{S}$ is able to frame an innocent $\mathcal{B}$ by reproducing and redistributing the watermarked content Y. Third, non-repudiation fails, even though B doesn't know W and cannot remove W from Y. A malicious $\mathcal{B}$ can deny his guilt by claiming that the pirated copy was created by $\mathcal{S}$ or a security breach in $\mathcal{S}$'s computing system. In fact, this attack weakens the security for both the buyer and the seller.

Attack Seller's Security

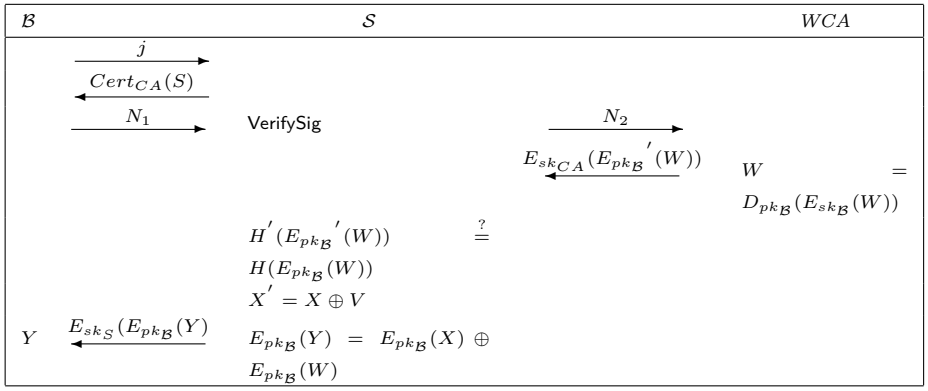
Besides the conspiracy attack explained above, a malicious buyer and the untrustworthy *WCA* could also collude. In this case, the *WCA* informs $\mathcal{B}$ the actual value of W directly, so that it is possible for $\mathcal{B}$ to remove his watermark from the watermarked digital content. Therefore, if the *WCA* is untrustworthy, the protocol fails to provide non-repudiation and hence security for the seller.

Failure of Probabilistic Cryptosystems

In the arbitration and identification protocol, the *WCA* is required by the judge $\mathcal{J}$ to decrypt $E_{pk_{WCA}}(W)$ and obtain the $\mathcal{B}$'s watermark W. Then $\mathcal{J}$ performs a validation on the correctness of $E_{pk_{\mathcal{B}'}}(W)$ sent by $\mathcal{S}$, by computing the encryption of W from the *WCA* with $\mathcal{B}$'s public key $pk_{\mathcal{B}}{}'$. If $E_{pk_{\mathcal{B}'}}(W)$ doesn't match $E_{pk_{\mathcal{B}'}}(W)$, then $\mathcal{J}$ announces the buyer is not guilty and the protocol halts. It is obvious that the verification won't work using probabilistic cryptosystems. The buyer-seller watermarking protocol requires watermarking insertion to be performed in the encrypted domain, and it should be achieved by employing privacy homomorphic cryptosystems. However, all efficient privacy homomorphic cryptosystems are probabilistic. As a result, the protocol fails to function properly as claimed.

3.2.2 Attacks on the Protocol of Ibrahim et al.

The players involved in Ibrahim et al.'s protocol [166] are the seller $\mathcal{S}$, the buyer $\mathcal{B}$, the certificate authority *CA*, and the judge $\mathcal{J}$. The protocol comprises two



Note: $N_1 = \{E_{pk_{\mathcal{B}}}(W), E_{sk_{\mathcal{B}}}(H(j)), E_{pk_{CA}}(E_{sk_{\mathcal{B}}}(W)), E_{sk_{\mathcal{B}}}(H(H(j)) + H(W)), Cert_{CA}(B)\}$
$N_2 = \{E_{pk_{CA}}(E_{sk_{\mathcal{B}}}(W)), Cert_{CA}(B)\}$

Figure 3.4. The watermark generation and insertion phase of Ibrahim et al.'s protocol

phases, namely the watermarking phase and the arbitration phase. The watermark generation and insertion protocol is reviewed in Figure 3.4.

Attack Seller's Security

In the watermarking phase, $\mathcal{B}$ generates $\mathcal{B}$'s secret watermark W, and W is approved by the CA. The watermarked content is $Y = X \oplus V \oplus W$, where V is $\mathcal{S}$'s watermark. Since $\mathcal{B}$ knows W, it is possible for $\mathcal{B}$ to remove his watermark W from the watermarked content Y. Hence, the protocol fails to provide non-repudiation and traitor traceability.

Ibrahim et al. assume that it is impossible for $\mathcal{B}$ to remove W from Y, because $\mathcal{B}$ doesn't have access of the original content X or the watermark embedding algorithm. Unfortunately, the assumption is unrealistic, and it can be combated by employing a blind watermarking scheme [187, 136], where the original content is not required to remove the watermark. On the other hand, there is no technical enforcement to ensure that B can't get the knowledge of the watermarking algorithm employed in the protocol. In fact, according to Kerckhoffs' principle [175] in cryptography, "a cryptosystem should be secure even if everything about the system, except the key, is public knowledge. The system must not require secrecy and can be stolen by the enemy without causing trouble." Therefore, the attack is effective, and this protocol provides neither the basic traceability nor the seller's security.

Failure of Probabilistic Cryptosystems

After $\mathcal{S}$ receives the encrypted value $E_{sk_{CA}}(E_{pk_{\mathcal{B}}}{}'(W))$ from CA, $\mathcal{S}$ decrypts $E_{sk_{CA}}(E_{pk_{\mathcal{B}}}{}'(W)$ using CA's public key pk_{CA}, and then computes the hash of the result $E_{pk_{\mathcal{B}}}{}'(W)$, i.e. $H'(E_{pk_{\mathcal{B}}}{}'(W))$. Next, $\mathcal{S}$ computes the hash of $E_{pk_{\mathcal{B}}}(W)$ sent earlier by $\mathcal{B}$, i.e. $H(E_{pk_{\mathcal{B}}}(W))$. $\mathcal{S}$ compares the values of $H'(E_{pk_{\mathcal{B}}}{}'(W))$ and $H(E_{pk_{\mathcal{B}}}(W))$. If they are equal the protocol continues, else the protocol throws exception and terminates. The protocol will fail with probabilistic cryptosystems, since $E_{pk_{\mathcal{B}}}{}'(W)$ computed by the CA and $E_{pk_{\mathcal{B}}}(W)$ provided by $\mathcal{B}$ would be different values. Following the same reason in Section 3.2.1, the protocol fails with homomorphic probabilistic cryptosystems.

Failure of Anonymity and Unlinkability

The protocol doesn't specify the registration phase. In each transaction, $\mathcal{B}$ provides $\mathcal{B}$'s PKI certificate $Cert_{CA}(B)$ issued by a trustworthy CA to $\mathcal{S}$. Since $Cert_{CA}(B)$ is not an anonymous certificate, $\mathcal{S}$ can identify $\mathcal{B}$. Therefore, it is clear that the protocol fails to provide $\mathcal{B}$'s anonymity and transaction unlinkability.

3.3 Cryptographic preliminaries

3.3.1 Group Signature Schemes

Group signature schemes [96] enable group members, each with his or her private signature key to produce signatures on behalf of the group. The scheme is called static if the identities of group members are fixed in the group setup phase, and is called dynamic if it allows adding and removing members to the group with time. Dynamic schemes have the advantage that instead of assigning a high level of trust to a single group manager, they provide more security with a lower level of trust by separating the group manager into an issuer, to issue private signature keys to the group members, and an opener, to open signatures. In the following, we recall the description of dynamic group signature schemes in [66].

The scenario consists of four kinds of parties: a trusted party for system setup, an authority called the issuer $\mathcal{I}$, an authority called the opener $\mathcal{O}$, and users $\mathcal{U}$ that may become group members. The communication between $\mathcal{I}$ and $\mathcal{U}$ takes place over private and authenticated channels.

The scheme consists of the algorithms GSgkg, GSukg, GSjoin, GSiss, GSsig, GSverify, GSopen, GSjudge. GSgkg outputs an issuer key isk, an opening key osk, and a group public key gpk on input a security parameter 1^k. GSukg outputs a user key

pair (upk, usk) on input a security parameter 1^k. GSjoin and GSiss are interactive algorithms run by $\mathcal{U}_i$ and $\mathcal{I}$ respectively. GSjoin receives (gpk, usk_i) as inputs and GSiss receives (gpk, isk, upk_i) as inputs. GSjoin outputs a private signing key gsk_i and GSiss outputs registration information reg_i to be stored in a registration table reg. GSsig outputs a signature s of a message m on input a secret key gsk. GSverify, on input a signature s, a message m and a group public key gpk, outputs a bit $b = 1$ if s is correct and $b = 0$ otherwise. GSopen, on input the group public key gpk, the registration table reg, an opening key osk, a message m and a signature s, outputs a pair $(i, proof)$, where i identifies the user $\mathcal{U}_i$ that computed s ($i = 0$ if no group member produced the signature) and $proof$ is a publicly verifiable proof that i computed s. GSjudge, on input a group public key gpk, an integer $i \geq 1$, a public key upk_i, a message m, a signature s and a proof $proof$, outputs $b = 1$ if $proof$ is a valid proof that i produced s and $b = 0$ otherwise.

A dynamic group signature scheme must provide the properties of anonymity, traceability and non-frameability. Anonymity requires that an adversary $\mathcal{A}$, unable to corrupt $\mathcal{O}$, cannot distinguish which of two signers of his choice signed a message of his choice. Traceability requires that $\mathcal{A}$, unable to corrupt $\mathcal{I}$ and $\mathcal{O}$ (albeit able to compromise osk), cannot compute a signature for which either an honest $\mathcal{O}$ cannot identify the user that produced it or cannot compute a proof $proof$ that a user $\mathcal{U}_i$ produced it. Non-frameability requires that $\mathcal{A}$ cannot produce a proof $proof$ that an honest user computed a valid signature unless the user indeed computed the signature. We refer to [66] for formal definitions, though the BSW protocols described in this chapter can be instantiated with any secure group signature scheme.

3.3.2 Homomorphic Cryptosystem

A homomorphic cryptosystem (or privacy homomorphism) refers to a cryptosystem E which is homomorphic with respect to some binary operators $\odot_{\mathcal{M}}$ in the plaintext space $\mathcal{M}$ and $\odot_{\mathcal{C}}$ in the ciphertext space $\mathcal{C}$, such that $\forall m_1, m_2 \in \mathcal{M} : E(m_1 \odot_{\mathcal{M}} m_2) = E(m_1) \odot_{\mathcal{C}} E(m_2)$. Homomorphic cryptosystems can be classified as two groups, namely the ones whose security relies on the *"decisional composite residuosity assumption" (DCRA)*, and the ones of the ElGamal class based on *"decisional Diffie-Hellman assumption" (DDH)*. Instead of *IND-CCA2*, the strongest security level a privacy homomorphism can reach is *IND-CPA*. Roughly speaking, indistinguishability under chosen plaintext attack [155] (IND-CPA) guarantees that an adversary does not get any knowledge about the plaintext m from the ciphertext c. For instance, the deterministic *RSA* cryptosystem [260] and the *ElGamal* cryptosystem [149] are multiplicative privacy homomorphism. In contrast to deterministic *RSA*, *ElGamal* is *IND-CPA*. The *Goldwasser-Micali* cryptosystem [155], the *Paillier* cryptosystem [230], and *Paillier*'s generalization the *Damgård-Jurik* cryptosystem [108] are additive privacy homomorphism.

Paillier cryptosystem

The public key homomorphic cryptosystem, proposed by Paillier [230] and generalization by Damgård and Jurik [108], is based on the problem of deciding whether a number is an N-th residue modulo N^2. This problem is believed to be computationally difficult and is linked to the hardness of factorization N, if N is the product of two large primes.

Let us now explain what an N-th residue is and how it can be used to encrypt data. Given the product of two large primes $N = pq$, the set $\mathbb{Z}_N$ of the integer numbers modulo N, and the set $\mathbb{Z}_N^*$ representing the integer numbers belonging to $\mathbb{Z}_N$ that are relatively prime with N, $z \in \mathbb{Z}_{N^2}^*$ is said to be an N-th residue modulo N^2 if there exists a number $y \in \mathbb{Z}_{N^2}^*$ such that $z = y^N \mod N^2$.

For a complete analysis of the Paillier cryptosystem one can refer to the original paper [230]. The set-up, encryption and decryption procedures are briefly reviewed in the following paragraphs.

Set-up Select p, q big primes. The private key is the least common multiple of $(p-1, q-1)$, denoted as $\lambda = lcm(p-1, q-1)$. Let $N = pq$ and g in $\mathbb{Z}_{N^2}^*$ an element of order[1] αN for some $\alpha \neq 0$ ($g = N + 1$ is usually a convenient choice). (N, g) is the public key.

Encryption Let $m < N$ be the plaintext, and $r < N$ a random value. The encryption c of m is:

$$c = E(m, r) = g^m r^N \mod N^2$$

Decryption Let $c < N^2$ be the ciphertext. The plaintext m hidden in c is:

$$m = D(c) = \frac{L(c^\lambda \mod N^2)}{L(g^\lambda \mod N^2)} \mod N$$

where $L(x) = \frac{x-1}{N}$. From the above equations, we can easily verify that the Paillier cryptosystem is additively homomorphic, since:

$$E(m_1, r_1) \cdot E(m_2, r_2) = g^{m_1+m_2}(r_1 r_2)^N = E(m_1 + m_2, r_1 r_2)$$

and

$$E(m, r)^a = (g^m (r)^N)^a = (g^{am}(r)^{aN}) = E(am, r^a).$$

3.3.3 Zero-Knowledge Proofs of Knowledge

A zero-knowledge proof of knowledge [64] is a two-party protocol between a prover and a verifier. The prover proves to the verifier knowledge of some secret input that

[1]The order of an integer a modulo N is the smallest positive integer k such that $a^k = 1 \mod N$.

fulfills some statement without disclosing this input to the verifier. The protocol should fulfill two properties. First, it should be a proof of knowledge, i.e., a prover without knowledge of the secret input convinces the verifier with negligible probability. More technically, there exists a knowledge extractor that extracts the secret input from a successful prover with all but negligible probability. Second, it should be zero-knowledge, i.e., the verifier does not learn any information about the secret input. More technically, for all possible verifiers there exists a simulator that, without knowledge of the secret input, yields a distribution that cannot be distinguished from the interaction with a real prover.

To express a zero-knowledge proof of knowledge, we follow the notation introduced by Camenisch and Stadler [89]. For example, $\mathsf{PK}\{(x) : y = f(x)\}$ denotes a "*zero-knowledge proof of knowledge of secret input x such that $y = f(x)$*". Letters in the parentheses, in this example x, denote the secret input, while y and the function f are also known to the verifier. The zero-knowledge proofs employed for the Type III BSW protocol will be elaborated in Section 3.7.

3.3.4 Verifiable Encryption

Verifiable encryption schemes enable the encrypter to ensure that the plaintext satisfies certain application-dependent properties without compromising secrecy. It can be employed in numerous applications including escrow schemes [296, 248], group signature and identity escrow schemes [56, 179], and digital payment with revocable anonymity [144, 87]. Specific schemes are proposed in [88] for both discrete-log- and factoring-based schemes. In the Type I and Type II BSW protocols described in Section 3.5 and Section 3.6, verifiable encryption is used for key escrow, such that the buyer can prove to the seller that the plaintext is valid without revealing any private information, and hence the buyer's privacy is preserved.

3.4 Security Definition of BSW Protocols

3.4.1 Blind Watermarking

A blind and readable watermarking scheme [60] consists of a setup algorithm WATsetup, a watermark embedding algorithm WATemb and finally a watermark detection algorithm WATdet. After an original content space $\mathcal{X}$ is determined, WATsetup outputs a secret watermarking key swk and a watermark space $\mathcal{W}$. WATemb(swk, X, W), on input swk, original content $X \in \mathcal{X}$, and watermark $W \in \mathcal{W}$, outputs watermarked content Y. The algorithm WATemb can be computed in the encrypted domain, where both W and the result Y are encrypted with

a public key of a public key encryption scheme. The algorithm $\mathsf{WATdet}(swk, Y)$ outputs the watermark W embedded in Y.

A secure watermarking scheme should be robust and collusion resistant. Let d be a distortion metric that quantifies the distortion suffered by a watermarked content Y when it underwent signal processing operations such as compression, filtering, noise addition, desynchronization, cropping, insertions, mosaicing, and collage. Let Y' be a distorted content. The robustness property requires that under a distortion metric d and a given distortion bound D, given swk output by $\mathsf{WATsetup}$ and Y output by $\mathsf{WATemb}(swk, X, W)$, a scheme is ϵ-robust if for every distorted content Y', $\mathsf{WATdet}(swk, Y')$ outputs W with overwhelming probability $1 - \epsilon$, if $d(Y, Y') \leq D$.

The collusion resistance property requires that a collusion of up to l parties cannot manipulate or remove the watermark from a watermarked content by comparing or composing their differently watermarked copies. In other words, it requires that under a distortion metric d and a given distortion bound D, a scheme is ϵ-secure against coalitions of size l, if all the p.p.t. adversaries (probabilistic polynomial time adversaries) win the game defined below with probability less than ϵ [278]. We formalize this property as follows:

Definition 1 (Collusion Resistant Watermarking)**.** *The collusion resistance property is defined through the following game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.*

- *Challenge. $\mathcal{C}$ runs $\mathsf{WATsetup}$ to get swk, picks random original content $X \in \mathcal{X}$, and, for $i = 1$ to l, picks random watermark $W_i \in \mathcal{W}$ and runs $Y_i = \mathsf{WATemb}(swk, X, W_i)$. $\mathcal{C}$ sends $(Y_1, \ldots, Y_l)$ to $\mathcal{A}$.*
- *Response. $\mathcal{A}$ outputs watermarked content Y'.*

$\mathcal{A}$ wins if $d(W_i, Y') \leq D$ and $\mathsf{WATdet}(swk, Y')$ outputs a watermark W' such that, for $i = 1$ to l, $W' \neq W_i$. A blind watermarking scheme is l collusion resistant if all p.p.t. adversaries (probabilistic polynomial time adversaries) $\mathcal{A}$ win the game above with negligible probability.

Current practical watermarking schemes do not provide collusion-resistance against any p.p.t. adversary. We assume that the watermarking scheme used to instantiate the protocol fulfills this definition, and thus we can prove that our protocol is secure against any p.p.t. adversary in Appendix B. When the protocol is instantiated with a given watermarking scheme, the security offered against malicious buyers is lowered to the security offered by the watermarking scheme.

3.4.2 Anonymous Buyer-Seller Watermarking Protocol

We define security following the ideal-world/real-world paradigm [90]. In the real world, a set of parties interact according to the protocol description in the presence of a real adversary $\mathcal{A}$, while in the ideal world dummy parties interact with an ideal functionality that carries out the desired task in the presence of an ideal adversary $\mathcal{E}$. A protocol ψ is secure if there is no environment $\mathcal{Z}$ that can distinguish whether it is interacting with adversary $\mathcal{A}$ and parties running protocol ψ or with the ideal process for carrying out the desired task, where ideal adversary $\mathcal{E}$ and dummy parties interact with an ideal functionality $\mathcal{F}_\psi$. More formally, we say that protocol ψ emulates the ideal process when, for any adversary $\mathcal{A}$, there exists a simulator $\mathcal{E}$ such that for all environments $\mathcal{Z}$, the ensembles $\text{IDEAL}_{\mathcal{F}_\psi,\mathcal{E},\mathcal{Z}}$ and $\text{REAL}_{\psi,\mathcal{A},\mathcal{Z}}$ are computationally indistinguishable. We refer to [90] for a description of how these ensembles are constructed. All functionality and every protocol invocation should be instantiated with a unique session-ID that distinguishes it from other instantiations. For the sake of ease of notation, we omit session-IDs from the description of our ideal functionalities.

We define an ideal functionality $\mathcal{F}_{\text{DRM}}$ that models the behavior and desirable properties of any copyright protection protocol in which buyers are provided with anonymity. We consider a setting with five parties: a seller $\mathcal{S}$ that sells protected digital content Y; a set of buyers $\mathcal{B}$ that purchase protected digital content from $\mathcal{S}$; a registration authority $\mathcal{R}$ where buyers must register before purchasing; a judge $\mathcal{J}$ that decides whether a buyer is guilty of releasing pirated copies; a deanonymization authority $\mathcal{D}$ that revokes the anonymity of a buyer when requested by $\mathcal{J}$. We assume that a group manager $\mathcal{GM}$ consists of three entities, namely a Trusted Registration Authority that generates keys for the involved parties, a registration authority $\mathcal{R}$, and a deanonymization authority $\mathcal{D}$. $\mathcal{F}_{\text{DRM}}$ is parameterized with a set of parties $\mathcal{P}$ that contains the aforementioned entities.

$\mathcal{F}_{\text{DRM}}$ models the properties that a copyright protection protocol should fulfill under three assumptions. First, the judge $\mathcal{J}$ is never corrupted by the ideal adversary $\mathcal{E}$. Second, parties can be corrupted statically, i.e., the ideal adversary $\mathcal{E}$ decides at the beginning of the protocol execution the set of parties it wishes to corrupt and cannot modify this set throughout the execution. Finally, $\mathcal{F}_{\text{DRM}}$ assumes that uncorrupted buyers never release pirated copies.

Under those assumptions, $\mathcal{F}_{\text{DRM}}$ requires that, when the seller is uncorrupted, buyers receive unique protected content Y at each purchase. This unique protected content, when released as a pirated copy, can be traced back to a single transaction. (In the case of a buyer-seller watermarking protocol, unique protected content is computed by embedding a different watermark at each purchase phase.) $\mathcal{F}_{\text{DRM}}$ also requires that, if the deanonymization authority $\mathcal{D}$ is uncorrupted, an uncorrupted seller is always able to get the identity of corrupted buyers that release pirated copies.

When the seller $\mathcal{S}$ is corrupted, $\mathcal{F}_{\text{DRM}}$ does not require buyers to receive the unique protected content Y. However, it requires that $\mathcal{S}$ is not able to frame uncorrupted buyers, who by assumption do not release pirated copies. Additionally, it requires that released pirated copies are traced back to corrupted buyers that collude with $\mathcal{S}$.

Below we formally describe $\mathcal{F}_{\text{DRM}}$. In Appendix B we prove that our Type III buyer-seller watermarking protocol (Section 3.7) *realizes* functionality $\mathcal{F}_{\text{DRM}}$. This means that our protocol fulfills the aforementioned properties.

Functionality $\mathcal{F}_{\text{DRM}}$

Parameterized with a set of parties $\mathcal{P}$, $\mathcal{F}_{\text{DRM}}$ works as follows, where $b = 1$ means the registration or deanonymization process succeeded and $d = 1$ means a content is not a pirated copy:

- Upon receiving (register) from buyer $\mathcal{B}_i$, $\mathcal{F}_{\text{DRM}}$ checks that $\mathcal{B}_i \in \mathcal{P}$. Then it sends $(\text{register}, \mathcal{B}_i)$ to $\mathcal{R}$. If $\mathcal{R}$ is corrupted, $\mathcal{F}_{\text{DRM}}$ receives a bit $(\text{regresp}, b)$ from the ideal adversary $\mathcal{E}$, else it sets $b = 1$. $\mathcal{F}_{\text{DRM}}$ sends $(\text{regresp}, b)$ to $\mathcal{B}_i$ and, if $b = 1$, includes $\mathcal{B}_i$ in its registration table T_{reg}.
- Upon receiving $(\text{request}, j)$ from buyer $\mathcal{B}_i$, where j identifies the item, $\mathcal{F}_{\text{DRM}}$ checks that $\mathcal{B}_i \in T_{\text{reg}}$. $\mathcal{F}_{\text{DRM}}$ sends $(\text{buyrequest}, j)$ to $\mathcal{S}$, who returns original content $(\text{reqresp}, X)$. $\mathcal{F}_{\text{DRM}}$ computes unique protected content Y from X. If $\mathcal{S}$ is corrupted, $\mathcal{F}_{\text{DRM}}$ receives (reqresp, Y') from $\mathcal{E}$ and sets Y to Y'. $\mathcal{F}_{\text{DRM}}$ sends (reqresp, Y) to $\mathcal{B}_i$ and stores $(Y, \mathcal{B}_i, d)$, where $d = 1$, in a transaction table T_{tra}.
- Upon receiving (release, Y) from $\mathcal{E}$, $\mathcal{F}_{\text{DRM}}$ sets d to 0 in the entry $(Y, \mathcal{B}_i, d)$ of T_{tra}. If no such entry exists, $\mathcal{F}_{\text{DRM}}$ stores $(Y, \mathcal{E}, 0)$ in T_{tra}.
- Upon receiving (detect, Y) from $\mathcal{S}$, if $d = 1$ in the entry $(Y, \mathcal{B}_i, d)$ or such entry does not exist, $\mathcal{F}_{\text{DRM}}$ sends $(\text{detresp}, \textit{not guilty})$ to $\mathcal{S}$ and $\mathcal{J}$. If $d = 0$, $\mathcal{F}_{\text{DRM}}$ sends $(\text{detect}, \mathcal{B}_i)$ to $\mathcal{D}$. If $\mathcal{D}$ is corrupted, $\mathcal{F}_{\text{DRM}}$ receives a bit $(\text{deanonym}, b)$ from $\mathcal{E}$, else sets $b = 1$. If $b = 0$, $\mathcal{F}_{\text{DRM}}$ sends $(\text{detresp}, \perp)$ to $\mathcal{S}$ and $\mathcal{J}$, and otherwise it sends $(\text{detresp}, \mathcal{B}_i, \textit{guilty})$ to $\mathcal{S}$ and $\mathcal{J}$.

In Appendix B we prove that our Type III buyer-seller watermarking protocol (Section 3.7) *realizes* functionality $\mathcal{F}_{\text{DRM}}$ in the $\mathcal{F}_{\text{REG}}$-hybrid model, where parties register their public keys at a Trusted Registration Authority and obtain from it

a common reference string. Do not confuse this entity with the registration entity $\mathcal{R}$. Below we depict the ideal functionality $\mathcal{F}_{\text{REG}}$. $\mathcal{F}_{\text{REG}}$ is parameterized with a distribution D and a set of participants $\mathcal{P}$, which is restricted to contain the registration authority $\mathcal{R}$, the deanonymization authority $\mathcal{D}$, the buyers $\mathcal{B}$, the seller $\mathcal{S}$ and the judge $\mathcal{J}$. $\mathcal{F}_{\text{REG}}$ can be implemented with a public key infrastructure.

Functionality $\mathcal{F}_{\text{REG}}$

Parameterized with a set of parties $\mathcal{P}$ and a distribution D, $\mathcal{F}_{\text{REG}}$ works as follows, where (crs) is a request of the common reference string, r is the common reference string, and v is the registered value such as P's public key:

- On input (crs) from party P, if $P \notin \mathcal{P}$ it aborts. Otherwise, if there is no value r recorded, it picks $r \leftarrow D$ and records r. It sends (crs, r) to P.
- Upon receiving $(\text{register}, v)$ from party $P \in \mathcal{P}$, it records the value (P, v).
- Upon receiving $(\text{retrieve}, P)$ from party $P' \in \mathcal{P}$, if (P, v) is recorded then return $(\text{retrieve}, P, v)$ to P'. Otherwise send $(\text{retrieve}, P, \perp)$ to P'.

3.5 Type I BSW protocol

3.5.1 Intuition Behind the Construction

The proposed buyer-seller watermarking (BSW) protocols are mainly based on two cryptographic primitives: group signatures and homomorphic encryption. Group signatures allow buyers to sign the purchase messages they send to the seller on behalf of the group of buyers. Thanks to that, the seller can verify the signature without knowing the buyer's identity, and thus purchases are anonymous. When a pirated copy is found and traced back to a particular purchase, the corresponding signature can be opened to know the identity of the buyer that released the pirated copy.

Homomorphic encryption allows the buyer and seller to jointly compute an encryption of the watermark to be embedded in the original content, in such a way that none of the parties knows the watermark. The encryption of the watermark is embedded in the encrypted domain, resulting in the encrypted version of the watermarked content to be delivered to the buyer. This is an essential property of

all BSW protocols. On the one hand, since the seller does not learn the watermark, later on a malicious seller cannot produce pirate copies that embed the watermark in order to frame an honest buyer. On the other hand, a malicious buyer can neither remove the watermark nor release pirate copies and claim that the seller has produced them.

The Type I BSW protocol involves four entities: a seller $\mathcal{S}$ that is the content provider and copyright holder, a buyer $\mathcal{B}$ that purchases a digital content from $\mathcal{S}$, a group manager $\mathcal{GM}$ that is a trustworthy authority, and a trustworthy judge $\mathcal{J}$ that adjudicates lawsuits against the infringement of copyrights. The protocol consists of three subprotocols. In the registration protocol, $\mathcal{GM}$ generates a group public key gpk, an issuing key isk and an opening key osk, and issues $\mathcal{B}$ a private signature key gsk before $\mathcal{B}$ purchases from $\mathcal{S}$.

In the watermark generation and embedding protocol, $\mathcal{B}$ and $\mathcal{S}$ engage in an electronic transaction for some digital content. First, $\mathcal{B}$ orders an item j, creates a group signature, and sends the encryption of $\mathcal{B}$'s secret watermark W to $\mathcal{S}$. $\mathcal{S}$ then performs the first round of watermarking embedding by inserting a unique watermark V to the original content X to get the watermarked content X'. The sole purpose of the watermark V is to enable $\mathcal{S}$ to identify a specific buyer from whose an illegal copy has been potentially generated. Next, $\mathcal{S}$ generates a random permutation function σ to permute the elements of $\mathcal{B}$'s encrypted watermark, and performs the second round of watermark embedding in the encrypted domain to insert the permuted watermark into the watermarked content X'. After the embedding, $\mathcal{S}$ sends the encrypted version of the watermarked content Y to $\mathcal{B}$, and $\mathcal{B}$ can do a decryption to retrieve the watermarked content Y.

In the identification and arbitration protocol, $\mathcal{S}$ receives a pirated copy Y and detects the watermark U in Y. $\mathcal{S}$ uses U to relate the pirated copy to a particular transaction and sends the stored transaction information to $\mathcal{J}$. $\mathcal{J}$ adjudicates if the accused buyer is guilty, and requests $\mathcal{GM}$ to open the group signature created by the buyer to recover the buyer's identity.

This approach takes care of both privacy protection for $\mathcal{B}$ and digital copyright protection for $\mathcal{S}$. $\mathcal{B}$ is only required to interact with the $\mathcal{GM}$ prior to transactions, and with $\mathcal{S}$ during transactions, hence no third party is involved. Furthermore, $\mathcal{B}$ is not required to participate in the copyright violator identification and arbitration process by providing $\mathcal{B}$'s private key or watermark. $\mathcal{J}$ is able to arbitrate the case and expose the identity of an adjudicated guilty buyer through the interactions with the $\mathcal{GM}$ and $\mathcal{S}$.

We define the following assumptions in the proposed scheme. Without these assumptions, the security property of the proposed scheme cannot be guaranteed. We assume a public key infrastructure PKI is available, such that each entity has a public and private key pair certified by the certification authority (CA). The group manager $\mathcal{GM}$ consists of a trusted party (also named as Trusted Registration

Authority) for group key generation, an issuer (also named as registration authority $\mathcal{R}$) for group member joining, and an opener (also named as deanonymization authority $\mathcal{D}$) for group signature opening. Additionally, the $\mathcal{GM}$ is assumed to be only party who knows the link between $\mathcal{B}$'s anonymous keys and identity. Furthermore, $\mathcal{J}$ is assumed to be a predetermined trustworthy entity that manages secret keys in the judicial system. For instance, in the protocol the buyer sends $\mathcal{J}$ the encrypted private key $C = E_{pk_{\mathcal{J}}}(sk_{\mathcal{B}}{}')$. Later in the arbitration phase, if some other judge is appointed to resolve the dispute, that judge may ask $\mathcal{J}$ to provide the necessary decryption of buyer's private key. For consistency, we assume that the digital content is a still image, although the protocol can be applied to other multimedia formats such as audio or video.

Justification for the Algorithm Selection

In the protocol, we need to employ an additive homomorphic encryption scheme that supports the following operation: on input two ciphertexts $\mathsf{Enc}(pk, x)$ and $\mathsf{Enc}(pk, y)$ that encrypt messages x and y, outputs a ciphertext $\mathsf{Enc}(pk, x + y) = \mathsf{Enc}(pk, x) \cdot \mathsf{Enc}(pk, y)$ that encrypts the addition of the messages. The public key homomorphic encryption schemes that support this operation include the exponential *ElGamal* cryptosystem [149], *Paillier* cryptosystem [230] and Paillier's variances. However, the exponential *ElGamal* cryptosystem requires to extract a discrete log at decryption, and this makes the cryptosystem only more efficient than Paillier cryptosystem or Paillier's variants when the plaintexts are small (e.g. less than 40 bits), such as for the design of e-voting schemes [104, 178]. In the watermark embedding phase of our protocol (to be explained in Section 3.5.3), the seller needs to sends an encryption of the whole image to the buyer. Take a standard 512×512 image for example, the seller needs to encrypt about 2M bits, which is of large computational complexity. To improve the protocol efficiency, instead of encrypting the image pixel-wise, an image is packed using the composite signal representation [115]. In such an approach, each encryption encodes several pixels, and the plaintext corresponding to each encryption will be of thousands of bits. For example, with 8 bits per pixel, an encryption encodes 128 pixels, and we have a 1024-bit plaintext. Therefore, the exponential *ElGamal* cryptosystem is not suitable for our setting. We choose to employ Paillier [230] cryptosystem or its variances, such as Damgård–Jurik cryptosystem [108], to instantiate the encryption scheme $(\mathsf{BKeygen}, \mathsf{BEnc}, \mathsf{BDec})$ employed in our protocol.

Moreover, the protocol needs to employ a group signature scheme to ensure the properties of non-frameability, anonymity and traceability. We try to keep our protocol generic so that any dynamic group signature scheme can be employed. As an example, the scheme proposed by Bellare et al. [66] is employed to instantiate the algorithms GSgkg, GSukg, GSjoin, GSiss, GSsig, $\mathsf{GSverify}$, GSopen, $\mathsf{GSjudge}$ of the group signature scheme.

In addition, our protocol incorporates a robust collusion-resistant watermarking scheme, such as the watermarking technique proposed by Cox et al. [101], and a verifiable encryption scheme, such as the scheme proposed by Camenisch et al. [88], for the key escrow of $\mathcal{B}$'s private key at $\mathcal{J}$ so that $\mathcal{B}$ can prove to $\mathcal{S}$ that the plaintext is valid without revealing $\mathcal{B}$'s private key.

Finally, the messages in the purchase phase between $\mathcal{B}$ and $\mathcal{S}$ are transferred over anonymous communication channels [132] to ensure anonymous outgoing connections. Anonymous channel provides privacy by preventing eavesdropping on the communication channel and disclosing the buyer's identity via opening the group signature it eavesdrops. Moreover, the messages in the registration phase and the arbitration phase are transferred over a secure (i.e. encrypted and authenticated) communication channel.

3.5.2 Type I Registration Protocol

The registration protocol, performed between the buyer $\mathcal{B}$ and the group manager $\mathcal{GM}$, is depicted in Figure 3.5.

1. In the group-key generation phase, $\mathcal{GM}$ generates a triple (gpk, osk, isk), with the group public key gpk to verify group signatures, the issuing key isk to issue signature keys for the group's users, and the opening key osk to open signatures.
2. In order to join the group, $\mathcal{B}$ generates a public and private key pair (upk_i, usk_i) and a signing and verification key pair (sk_B, pk_B). Then $\mathcal{B}$ runs the *user-key generation* algorithm GSukg to generate a public and private key pair (upk_i, usk_i).
3. $\mathcal{B}$ creates a signature sig_B on pk_B using usk_i, and sends sig_B and pk_B to $\mathcal{GM}$. $\mathcal{GM}$ verifies $\mathcal{B}$'s signature sig_B, issues $\mathcal{B}$ a certificate $cert_B$, and stores (sig_B, pk_B) in $\mathcal{GM}$'s registration table reg. After receiving the certificate $cert_B$ from $\mathcal{GM}$, $\mathcal{B}$ generates the private group signature key gsk from the tuple $(\mathcal{B}_i, pk_B, sk_B, cert_B)$.

3.5.3 Type I Watermark Generation and Embedding Protocol

The watermark generation and embedding protocol can be executed multiple times for multiple transactions between the seller $\mathcal{S}$ and the buyer $\mathcal{B}$. The exact algorithm depends on the underlying watermarking techniques. $\mathcal{S}$ and $\mathcal{B}$ first need to negotiate a purchase agreement j on rights and specification of the digital content X. Hence j uniquely binds a particular transaction to the item of interest

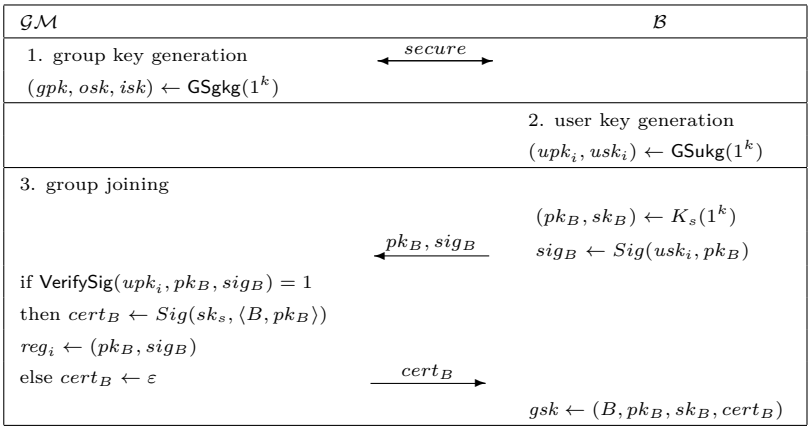


Figure 3.5. The registration protocol of the Type I BSW protocol performed between $\mathcal{B}$ and $\mathcal{GM}$

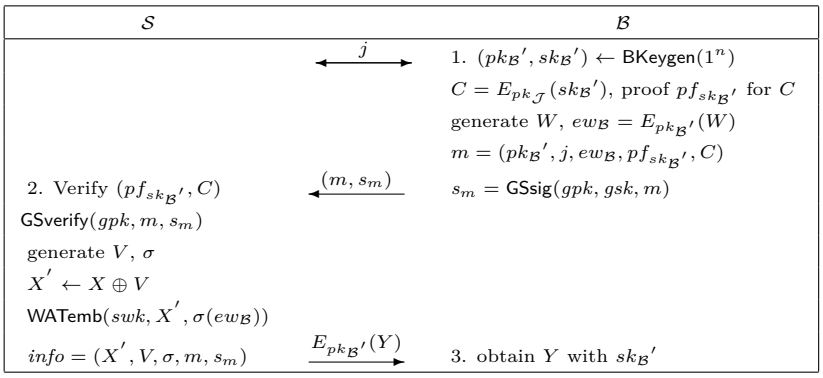


Figure 3.6. The watermark generation and embedding protocol of the Type I BSW protocol performed between $\mathcal{S}$ and $\mathcal{B}$

X. All messages are transferred over a secure and anonymous communication channel. Figure 3.6 depicts the protocol with the following steps:

1. $\mathcal{B}$ generates a one-time anonymous public and private key pair $(pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}')$ on input 1^n, where n is a security parameter. $\mathcal{B}$ computes an escrow cipher $C = E_{pk_{\mathcal{J}}}(sk_{\mathcal{B}}{}')$ to recover $\mathcal{B}$'s private key $sk_{\mathcal{B}}{}'$ from the $\mathcal{GM}$ in case of disputes. The verifiable proof $pf_{sk_{\mathcal{B}}{}'}$ for C is to assure $\mathcal{S}$ that the encrypted message is valid without compromising $sk_{\mathcal{B}}{}'$ to $\mathcal{S}$. In each transaction, $\mathcal{B}$ generates $\mathcal{B}$'s secret watermark W and encrypts W using $pk_{\mathcal{B}}{}'$, as $ew_{\mathcal{B}} = E_{pk_{\mathcal{B}}{}'}(W)$. For the message $m = (pk_{\mathcal{B}}{}', j, ew_{\mathcal{B}}, pf_{sk_{\mathcal{B}}{}'}, C)$, $\mathcal{B}$ runs the *group-sign* algorithm GSsig to create a group signature s_m to m using $\mathcal{B}$'s group signature key gsk, as $s_m = \mathsf{GSsig}(gpk, gsk, m)$. Then $\mathcal{B}$ sends (m, s_m) to $\mathcal{S}$.

2. $\mathcal{S}$ first verifies $\mathcal{B}$'s the verifiable proof $pf_{sk_{\mathcal{B}}{}'}$, and executes the *group signature verification* algorithm to verify $\mathcal{B}$'s group signature s_m with the group public key gpk, as $\mathsf{GSverify}(gpk, m, s_m)$. If the verification fails, the protocol halts. Otherwise, $\mathcal{S}$ generates a unique watermark V in compliance with the features of X and embeds V to X, performs the bit-wise watermark embedding $\mathsf{WATemb}(swk, X, V)$ to obtain the intermediate watermarked content, as

$$X^{'} = X \oplus V = \{x_1 \oplus v_1, x_2 \oplus v_2, ..., x_n \oplus v_n\} \tag{3.1}$$

where $\oplus$ denotes the watermark embedding operation in the plaintext domain.

Next, $\mathcal{S}$ generates a random permutation function σ to permute the elements of the encrypted W. The bit-wise encryption of $\mathcal{B}$'s watermark W can be written as

$$\begin{aligned} E_{pk_{\mathcal{B}}{}'}(W) &= E_{pk_{\mathcal{B}}{}'}(\{w_1, w_2, ..., w_n\}) \\ &= \{E_{pk_{\mathcal{B}}{}'}(w_1), E_{pk_{\mathcal{B}}{}'}(w_2), ..., E_{pk_{\mathcal{B}}{}'}(w_n)\}, \end{aligned} \tag{3.2}$$

$\mathcal{S}$ computes permuted watermark encryption as

$$\sigma(E_{pk_{\mathcal{B}}{}'}(W)) = E_{pk_{\mathcal{B}}{}'}(\sigma(W)). \tag{3.3}$$

$\mathcal{S}$ then performs the second watermark embedding in the encrypted domain $\mathsf{WATemb}(swk, X', \sigma(ew_{\mathcal{B}}))$ by applying homomorphic cryptosystem as

$$\begin{aligned} E_{pk_{\mathcal{B}}{}'}(Y) &= E_{pk_{\mathcal{B}}{}'}(X^{'}) \otimes E_{pk_{\mathcal{B}}{}'}(\sigma(W)) \\ &= E_{pk_{\mathcal{B}}{}'}(X^{'} \oplus \sigma(W)), \end{aligned} \tag{3.4}$$

where $\otimes$ denotes the watermark embedding operation in the encrypted domain.

Note that the above computation is possible because we assume the encryption $E_{pk_{\mathcal{B}'}}(\cdot)$ is homomorphic with respect to the watermark embedding operation $\oplus$. $\mathcal{S}$ stores $info = (X^{'}, V, \sigma, m, s_m)$ in the transaction table as the record, and delivers the encrypted content of Y to $\mathcal{B}$.

3. $\mathcal{B}$ obtains the watermarked content Y by decryption $D_{sk_{\mathcal{B}'}} E_{pk_{\mathcal{B}'}}(Y)$.

3.5.4 Type I Identification and Arbitration Protocol

The identification and arbitration protocol, executed among the seller $\mathcal{S}$, the judge $\mathcal{J}$, and the $\mathcal{GM}$, is depicted in Figure 3.7.

1. Once a pirated copy Y of X is found, $\mathcal{S}$ runs the watermark detection algorithm $\mathsf{WATdet}(swk, Y)$ and extracts the watermark U from Y, where swk is the watermarking secret key. $\mathcal{S}$ correlates U with every V in the transaction table to choose the record $info$ with the highest correlation. In addition, $\mathcal{S}$ computes the encryption of the secret watermarking key swk for $\mathcal{J}$, as $ck = E_{pk_{\mathcal{J}}}(swk)$. It is to enable $\mathcal{GM}$ to perform the watermark detection using the same secret watermarking key as $\mathcal{S}$. Then $\mathcal{S}$ sends the unauthorized copy Y, the transaction record $info$, and the watermarking secret key ck to $\mathcal{J}$.

2. Upon receiving the message from $\mathcal{S}$, $\mathcal{J}$ parses $info$ as $(X^{'}, V, \sigma, m, s_m)$, and m as $(pk_{\mathcal{B}}{}', j, ew_{\mathcal{B}}, pf_{sk_{\mathcal{B}'}}, C)$ where $s_m = \mathsf{GSsig}(gpk, gsk, m)$. $\mathcal{J}$ verifies the group signature $\mathsf{GSverify}(gpk, m, s_m)$ and performs a number of decryptions to obtain $\mathcal{B}$'s private key $sk_{\mathcal{B}}{}' = D_{sk_{\mathcal{J}}}(C)$, the secret watermarking key $swk = D_{pk_{\mathcal{J}}}(ck)$, and $\mathcal{B}$'s secret watermark $W = D_{sk_{\mathcal{B}'}}(ew_{\mathcal{B}})$. Next, $\mathcal{J}$ permutes the watermark W as $\sigma(W)$, and runs the watermark detection algorithm $\mathsf{WATdet}(swk, Y)$ and checks if $\sigma(W)$ indeed presents in the content X'. If $\sigma(W)$ is found, the suspected buyer is guilty, and $\mathcal{J}$ sends a court order $M_{open} = (m, s_m)$ to $\mathcal{GM}$. Otherwise, $\mathcal{J}$ informs $\mathcal{S}$ that the buyer is innocent and the protocol halts. Note that until now, the buyer has been anonymous.

3. To recover $\mathcal{B}$'s identity, $\mathcal{GM}$ accesses the registration table reg and runs the *group signature open* algorithm $\mathsf{GSopen}(gpk, osk, reg, pk_{\mathcal{B}}{}', s_m)$ using its opening key osk and its registration table reg, and obtains the identity $\mathcal{B}_i$ and a claim proof τ.

4. $\mathcal{J}$ verifies $\mathcal{GM}$'s claim by running the *group signature judge algorithm* $\mathsf{GSjudge}(gpk, \mathcal{B}_i, upk_i, pk_{\mathcal{B}}{}', s_m, \tau)$. If the $\mathcal{GM}$'s claim is verified, $\mathcal{J}$ closes the

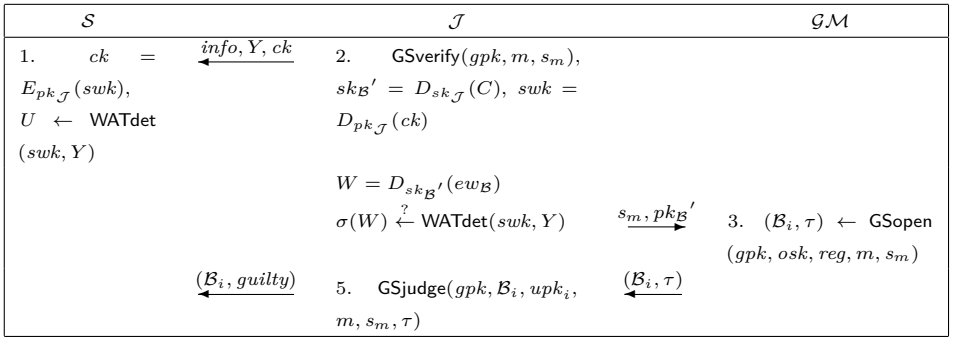


Figure 3.7. The identification and arbitration protocol of the Type I BSW protocol performed among $\mathcal{S}$, $\mathcal{J}$, and $\mathcal{GM}$

case and adjudicates that the buyer whose identity is $\mathcal{B}_i$ is guilty. Otherwise, $\mathcal{J}$ informs $\mathcal{S}$ with an empty string ($\perp$) and the protocol halts.

3.6 Type II BSW protocol

3.6.1 Intuition Behind the Construction

The Type II BSW protocol has the similar design principle as that of the Type I BSW protocol (in Section 3.5). The protocol also involves the seller $\mathcal{S}$, the buyer $\mathcal{B}$, the trustworthy group manager $\mathcal{GM}$, and a trustworthy judge $\mathcal{J}$; and it consists of three phases, namely registration, watermarking, and arbitration.

The essential difference between the Type II and Type I BSW protocols lies in the watermark generation and embedding. In the watermarking phase, $\mathcal{B}$ and $\mathcal{S}$ both contribute their secret watermarks $W_{\mathcal{B}}$ and $W_{\mathcal{S}}$. $\mathcal{B}$ first sends the encryption of $W_{\mathcal{B}}$ to $\mathcal{S}$, and $\mathcal{S}$ computes the encryption of a composite watermark W based on the encryptions of $W_{\mathcal{B}}$ and $W_{\mathcal{S}}$ by using homomorphic encryptions. In addition, $\mathcal{S}$ needs to perform a double watermark embedding. $\mathcal{S}$ first generate a unique watermark V with the only purpose to identify a transaction record in $\mathcal{S}$'s transaction table later on, and $\mathcal{S}$ embeds V to the original content X in the plaintext domain to obtain an intermediate watermarked content X'. Hereafter, $\mathcal{S}$ embeds the composite watermark W to X' in the encrypted domain by using homomorphic encryptions to obtain the encrypted version of the final watermarked content Y. After receiving the encryption of Y from $\mathcal{S}$, $\mathcal{B}$ performs decryption and obtains Y.

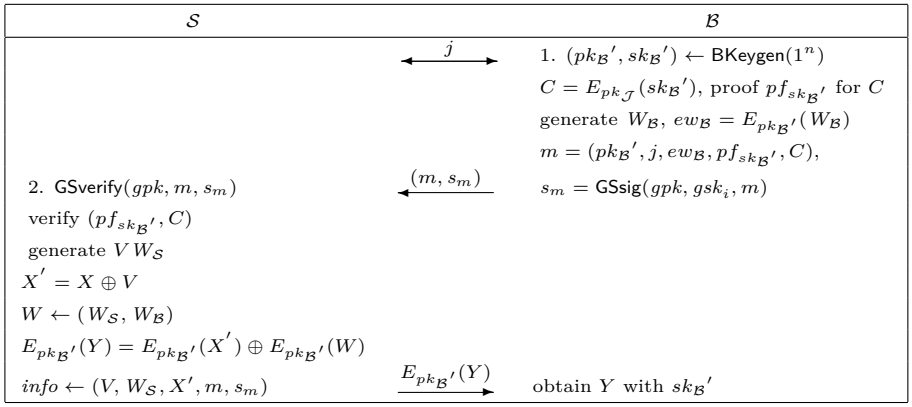


Figure 3.8. The watermark generation and embedding protocol of the Type II BSW protocol performed between the seller $\mathcal{S}$ and the buyer $\mathcal{B}$

In this scheme, neither $\mathcal{B}$ nor $\mathcal{S}$ knows the composite watermark W. $\mathcal{B}$ doesn't learn anything about $\mathcal{S}$'s watermark $W_{\mathcal{S}}$, the original content X and the index watermark V. Besides, $\mathcal{B}$'s watermark $W_{\mathcal{B}}$ and the final watermarked content Y remains unknown to $\mathcal{S}$. Subsequently, the watermark detections by the judge $\mathcal{J}$ in the arbitration phase needs to be adjusted to accommodate the modifications in the watermarking phase. Apart from that, the assumptions of the Type I BSW protocols still hold for the Type II BSW protocols.

The justification for the algorithm selection of the Type II BSW protocol is the same as what has been explained in Section 3.5.1.

3.6.2 Type II Registration Protocol

In the registration phase, the buyer $\mathcal{B}$ registers at the group manager $\mathcal{GM}$ with $\mathcal{B}$'s public and private key pair (upk_i, usk_i), the signing and verification key pair (sk_B, pk_B) and $\mathcal{B}$'s identity $\mathcal{B}_i$. In return, $\mathcal{GM}$ issues $\mathcal{B}$ a private group signature key gsk and stores $\mathcal{B}$'s information in $\mathcal{GM}$'s registration table reg. The data flow of the Type II BSW registration protocol is the same of the one in the Type I BSW protocol, which is depicted in Figure 3.5.

3.6.3 Type II Watermark Generation and Embedding Protocol

The watermark generation and insertion protocol is executed between the seller $\mathcal{S}$ and the buyer $\mathcal{B}$, and it is depicted in Figure 3.8.

1. $\mathcal{B}$ first orders an item j from $\mathcal{S}$, and j uniquely bind a particular transaction to the item of interest X. $\mathcal{B}$ generates a one-time anonymous public and private key pair $(pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}')$. For key escrow, $\mathcal{B}$ encrypts the secret key $sk_{\mathcal{B}}{}'$ with $\mathcal{J}$'s encryption key $pk_{\mathcal{J}}$, and computes a verifiable proof $pf_{sk_{\mathcal{B}}{}'}$ for the escrow cipher C to assure $\mathcal{S}$ that the encrypted message is valid without compromising $sk_{\mathcal{B}}{}'$. For each transaction, $\mathcal{B}$ generates a unique watermark $W_{\mathcal{B}}$, and sends the encrypted $W_{\mathcal{B}}$ and the other public information as a message $m = (pk_{\mathcal{B}}{}', j, ew_{\mathcal{B}}, pf_{sk_{\mathcal{B}}{}'}, C)$ to $\mathcal{S}$. Besides, $\mathcal{B}$ also creates a group signature of m with the private signature key gsk as $s_m = \mathsf{GSsig}(gpk, gsk_i, m)$, and sends the signature s_m to $\mathcal{S}$.

2. $\mathcal{S}$ verifies $\mathcal{B}$'s verifiable proof $pf_{sk_{\mathcal{B}}{}'}$ and the group signature s_m, and generates two unique watermarks V and $W_{\mathcal{S}}$ for each transaction. The first round of watermark insertion $\mathsf{WATemb}(swk, X, V)$ is performed as:

$$X^{'} = X \oplus V, \tag{3.5}$$

where $\oplus$ the watermark embedding operation in the message space. Note that the only purpose of V is to search the sales record in case $\mathcal{S}$ finds a pirated copy of her products [203, 189].

3. $\mathcal{S}$ computes the composite watermark W in the encrypted domain using additive homomorphic encryptions:

$$\begin{aligned} E_{pk_{\mathcal{B}}{}'}(W) &= E_{pk_{\mathcal{B}}{}'}(W_{\mathcal{S}}) \times E_{pk_{\mathcal{B}}{}'}(W_{\mathcal{B}}) \\ &= E_{pk_{\mathcal{B}}{}'}(W_{\mathcal{S}} + W_{\mathcal{B}}), \end{aligned} \tag{3.6}$$

where $+$ denotes addition and $\times$ denotes multiplication defined in the Galois field, respectively.

4. $\mathcal{S}$ performs the second round of watermark insertion in the encrypted domain $\mathsf{WATemb}(swk, X^{'}, E_{pk_{\mathcal{B}}{}'}(W))$:

$$\begin{aligned} E_{pk_{\mathcal{B}}{}'}(Y) &= E_{pk_{\mathcal{B}}{}'}(X^{'}) \otimes E_{pk_{\mathcal{B}}{}'}(W) \\ &= E_{pk_{\mathcal{B}}{}'}(X^{'} \oplus W), \end{aligned} \tag{3.7}$$

where $\otimes$ denotes the corresponding operation in the encrypted domain. This computation is possible because we assume the encryption $E_{pk_{\mathcal{B}}{}'}(\cdot)$ is privacy homomorphic with respect to $\oplus$. $\mathcal{S}$ then stores $(V, W_{\mathcal{S}}, X', m, s_m)$ as *info* in the transaction table, and delivers the encrypted content Y to $\mathcal{B}$.

5. After decryption $D_{sk_{\mathcal{B}}{}'}E_{pk_{\mathcal{B}}{}'}(Y)$, $\mathcal{B}$ obtains the watermarked content Y from $\mathcal{S}$.

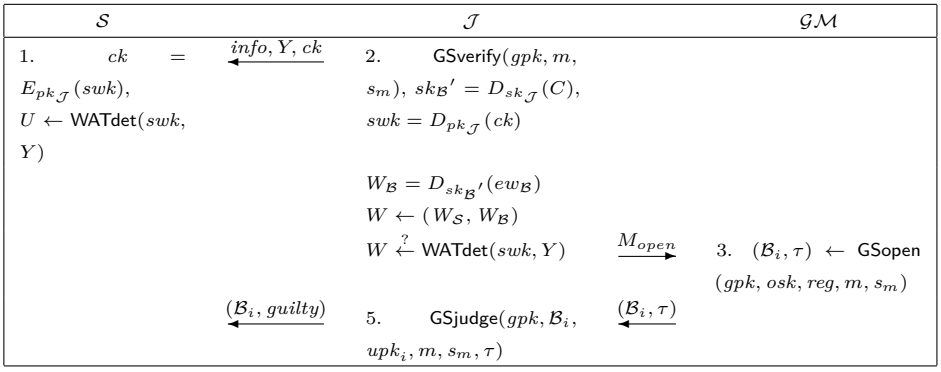


Figure 3.9. The copyright violator identification and arbitration protocol of the Type II BSW protocol performed among the seller $\mathcal{S}$, the judge $\mathcal{J}$, and $\mathcal{GM}$

3.6.4 Type II Identification and Arbitration Protocol

The identification and arbitration protocol is executed among the seller $\mathcal{S}$, a judge $\mathcal{J}$, and $\mathcal{GM}$, as depicted in Figure 3.9.

1. In case $\mathcal{S}$ finds a pirated copy Y, $\mathcal{S}$ extracts the watermark U from Y, and searches the sales record by correlating U with every V in $\mathcal{S}$'s table. Then $\mathcal{S}$ provides the encryption of the secret watermarking key $ck = E_{pk_{\mathcal{J}}}(swk)$ and $\mathcal{B}$'s relevant information *info* together with the unauthorized copy Y to $\mathcal{J}$.

2. Upon receiving the message from $\mathcal{S}$, $\mathcal{J}$ parses *info* as $(V, W_{\mathcal{S}}, X', m, s_m)$ and m as $(pk_{\mathcal{B}}{}', j, ew_{\mathcal{B}}, pf_{sk_{\mathcal{B}}{}'}, C)$. $\mathcal{J}$ first verifies the group signature $\mathsf{GSverify}(gpk, m, s_m)$, and aborts if the signature is not verified. Otherwise, $\mathcal{J}$ recovers $\mathcal{B}$'s private key and the secret watermarking key swk from the decryptions $sk_{\mathcal{B}}{}' = D_{sk_{\mathcal{J}}}(E_{pk_{\mathcal{J}}}(sk_{\mathcal{B}}{}'))$ and $swk = D_{pk_{\mathcal{J}}}(ck)$, respectively, and further obtains $\mathcal{B}$'s secret watermark as $W_{\mathcal{B}} = D_{sk_{\mathcal{B}}{}'}(ew_{\mathcal{B}})$. Then $\mathcal{J}$ computes $W = W_{\mathcal{B}} + W_{\mathcal{S}}$. Then $\mathcal{J}$ performs the watermark detection and checks if W indeed presents in the pirated copy Y. If W is found in Y, $\mathcal{J}$ sends a court order $M_{open} = (m, s_m)$ to $\mathcal{GM}$. Otherwise $\mathcal{J}$ sends (*not guilty*) to $\mathcal{S}$, and the protocol halts. Note that, the buyer's identity is not exposed until now.

3. To recover the buyer's identity, $\mathcal{GM}$ opens the buyer's group signature executes the *group signature open* algorithm $\mathsf{GSopen}(gpk, osk, reg, m, s_m)$, to obtain the identity $\mathcal{B}_i$ and a proof τ.

4. Upon receiving the recovered identity $\mathcal{B}_i$ and a claim proof τ, $\mathcal{J}$ verifies $\mathcal{GM}$'s claim. If the claim is verified, $\mathcal{J}$ closes the case and informs $\mathcal{S}$ that the buyer with identity $\mathcal{B}_i$ is guilty.

3.7 Type III BSW protocol

3.7.1 Intuition Behind the Construction

The Type III BSW protocol improves the Type I and Type II BSW protocols in a number of aspects. Primarily, the double watermarking embedding from the Type I and Type II BSW protocols is avoided. Double watermark insertions may cause a degradation of the final quality of the distributed content. When applied independently, the second watermark could confuse or discredit the authority of the first watermark, thus acting as an actual "ambiguity attack" [105]. That is avoided by first computing a composite watermark W, which is composed of the buyer's secret watermark $W_{\mathcal{B}}$, the seller's secret watermark $W_{\mathcal{S}}$, and a transaction index ϕ.

Moreover, the protocol consists of four phases: setup, registration, purchase, and arbitration. The involved parties are a buyer $\mathcal{B}$, a seller $\mathcal{S}$, a judge $\mathcal{J}$, a registration authority $\mathcal{R}$, and a deanonymization authority $\mathcal{D}$. It is assumed that the registration authority $\mathcal{R}$, the deanonymization authority $\mathcal{D}$, together with the judge $\mathcal{J}$ are trustworthy.

In the setup phase, a Trusted Registration Authority releases the group public key gpk, gives the issuer secret key isk to the registration authority $\mathcal{R}$ and the opener secret key osk to the deanonymization authority $\mathcal{D}$. $\mathcal{R}$ acts as the issuer of the group signature scheme, and $\mathcal{D}$ as the opener. Additionally, buyers register their public keys at the Trusted Registration Authority, and $\mathcal{J}$ also registers a public key.

In the registration phase, buyers query $\mathcal{R}$ and obtain a private signing key gsk of the group signature scheme. $\mathcal{R}$ obtains registration information reg.

In the purchase phase, a buyer $\mathcal{B}$ requests the item j from $\mathcal{S}$, randomly picks a watermark $W_{\mathcal{B}}$ and sends the bitwise encryption of $W_{\mathcal{B}}$ to $\mathcal{S}$. Then, $\mathcal{B}$ computes a group signature s_m on the request message m, sends it to $\mathcal{S}$ and proves in zero-knowledge that the request is correctly computed. $\mathcal{S}$ randomly picks a unique index ϕ and $\mathcal{S}$'s watermark $W_{\mathcal{S}}$, and computes the encryption of the composite watermark w by using the homomorphic property of the encryption scheme such that $W = \phi||(W_{\mathcal{S}} \oplus W_{\mathcal{B}})$, where $\phi \in \{0,1\}^{l_1}$ and $W_{\mathcal{S}} \in \{0,1\}^{l_2}$ are chosen by the seller, while $W_{\mathcal{B}} \in \{0,1\}^{l_2}$ is chosen by buyer. $\mathcal{S}$ embeds W in the original content X in the encrypted domain and sends the encryption of the watermarked content

to $\mathcal{B}$. $\mathcal{B}$ then decrypts the message to obtain the watermarked content Y_j from $\mathcal{S}$. In such a way, the double watermark embedding is avoided. None of the parties knows the composite watermark W. $\mathcal{S}$ does not learn $\mathcal{B}$'s watermark $W_{\mathcal{B}}$ and the final watermarked content Y_j, and $\mathcal{B}$ doesn't know $\mathcal{S}$'s watermark $W_{\mathcal{S}}$, the index ϕ and the original content X.

Justification for the Algorithm Selection

The third improvement of the Type III BSW protocol is to incorporate zero-knowledge proofs of knowledge in the watermarking phase. A proof of knowledge $\mathsf{PK}\{(sk') : (pk', sk') \leftarrow \mathsf{BKeygen}(1^k) \wedge C \leftarrow \mathsf{Enc}(pk, sk')\}$ is employed, i.e., a proof that C is a correct encryption under pk of the secret key sk' related with public key pk', so that a party in possession of the secret key sk related with pk can recover sk' from C. Two candidate schemes can be employed to instantiate the encryption scheme $(\mathsf{JKeygen}, \mathsf{JEnc}, \mathsf{JDec})$ used in the construction in Section 3.7.2, namely the verifiable encryption by Camenisch et al. [88] and the fair encryption of RSA keys by Poupard and Stern [248]. Despite the claim of Camenisch et al. [88] that Poupard and Stern's solution [248] may overlook the fact that the underlying encryption scheme provides security against chosen ciphertext attacks, we decide to employ Poupard and Stern's scheme due to its efficiency of zero knowledge proofs. Besides, a proof of knowledge of the statement $\mathsf{PK}\{(b) : c \leftarrow \mathsf{Enc}(pk, b) \wedge b \in \{0, 1\}\}$ is also employed, i.e., a proof that the value b encrypted in ciphertext c under public key pk is a bit. Such a proof is described in [108].

We employ a public key homomorphic encryption scheme that supports two operations. An operation $\odot$ that, on input two ciphertexts $\mathsf{Enc}(pk, x)$ and $\mathsf{Enc}(pk, y)$ that encrypt messages x and y, outputs a ciphertext $\mathsf{Enc}(pk, x + y) = \mathsf{Enc}(pk, x) \odot \mathsf{Enc}(pk, y)$ that encrypts the addition of the messages, and an operation $\otimes$ that, on input a message x and a ciphertext $\mathsf{Enc}(pk, y)$, outputs a ciphertext $\mathsf{Enc}(pk, xy) = x \otimes \mathsf{Enc}(pk, y)$ that encrypts the multiplication of the messages x and y. Following the same reason explained in Section 3.5.1, the public key homomorphic encryption scheme proposed by Paillier [230] and its generalization by Damgård and Jurik [108] support these operations, and therefore can be used to instantiate the encryption scheme $(\mathsf{BKeygen}, \mathsf{BEnc}, \mathsf{BDec})$ employed in Section 3.7.2. In the protocol, we need a function that, on input bit b and an encryption $\mathsf{Enc}(pk, b')$ of bit b', computes the encryption $\mathsf{Enc}(pk, b \oplus b')$, where $\oplus$ denotes the exclusive or operation. With Paillier encryption, the function can be computed as follows. If $b = 0$, output $\mathsf{Enc}(pk, b')$. If $b = 1$, output $\mathsf{Enc}(pk, b) \odot (-1 \otimes \mathsf{Enc}(pk, b'))$.

There are a number of assumptions for the Type III BSW protocol. We require a Trusted Registration Authority (e.g. a public key infrastructure) that certifies public keys of the parties. It is assumed that the judge $\mathcal{J}$ and the Trusted

Registration Authority (do not confuse this entity with $\mathcal{R}$) are trustworthy. For consistency and ease of explanation, the digital content is assumed to be an image, although the protocol can be applied to other multimedia formats. In addition, the protocol is suitable to employ any types of group signature schemes, and we follow the group signature construction formalized in Section 3.3. Finally, the messages in the purchase phase are transferred over anonymous communication channels [132] to ensure anonymous outgoing connections between $\mathcal{B}$ and $\mathcal{S}$. Anonymous channel provides privacy, which prevents a malicious opener that eavesdrops the communication channel from deanonymizing the buyer that is requesting an item (by opening the group signature it eavesdrops). The messages in the registration phase and the arbitration phase are transferred over a secure (i.e. encrypted and authenticated) communication channel.

3.7.2 Type III Protocol Construction

The security definition of BSW protocols is provided in Section 3.4. This section specifies the construction of the proposed Type III BSW protocol. In the following, $(\mathsf{JKeygen}, \mathsf{JEnc}, \mathsf{JDec})$ and $(\mathsf{BKeygen}, \mathsf{BEnc}, \mathsf{BDec})$ stand for the algorithms for key generation, encryption and decryption of the public key encryption schemes used by $\mathcal{J}$ and $\mathcal{B}$ respectively. They are described in Section 3.3.

In the setup phase, the trusted registration functionality $\mathcal{F}_{\text{REG}}$ runs the setup algorithm GSgkg of the group signature scheme, stores the group public key gpk and sends the issuer's secret key isk to $\mathcal{R}$ and the opening secret key osk to $\mathcal{D}$. Every party can obtain gpk by sending (crs) to $\mathcal{F}_{\text{REG}}$.

Additionally, each buyer $\mathcal{B}_i$ runs GSukg to obtain a user key pair (upk_i, usk_i) and registers upk_i at $\mathcal{F}_{\text{REG}}$. The judge $\mathcal{J}$ runs his key generation algorithm $\mathsf{JKeygen}$ in order to generate a key pair $(pk_{\mathcal{J}}, sk_{\mathcal{J}})$ and registers $pk_{\mathcal{J}}$ at $\mathcal{F}_{\text{REG}}$. Every party can retrieve public keys of other parties by querying $\mathcal{F}_{\text{REG}}$.

Finally, the seller $\mathcal{S}$ executes the watermarking setup algorithm $\mathsf{WATsetup}$ to obtain secret watermarking key swk. $\mathcal{S}$ encrypts $ck = \mathsf{JEnc}(pk_{\mathcal{J}}, swk)$ and sends ck to $\mathcal{J}$.

After the setup phase, our protocol consists of three phases: registration, purchase, and arbitration. We begin with a high level description of our construction. Details of the algorithms can be found below.

Protocol BSW

- **Registration.** When $\mathcal{B}_i$ is activated with (register), $\mathcal{B}_i$ and $\mathcal{R}$ execute GSjoin and GSiss respectively. $\mathcal{B}_i$ inputs (gpk, usk_i), and $\mathcal{R}$ inputs

(gpk, isk, upk_i). $\mathcal{B}_i$ obtains a private signing key gsk_i and outputs (regresp, 1), while $\mathcal{I}$ obtains registration information reg_i to be stored in the registration table reg.

- **Purchase.** When $\mathcal{B}_i$ is activated with (request, j) and $\mathcal{S}$ is activated with (reqresp, X), $\mathcal{B}_i$ and $\mathcal{S}$ run the interactive algorithms $\mathsf{Request}$ and $\mathsf{Response}$ respectively. $\mathcal{B}_i$ inputs the group public key gpk, her private signing key gsk_i, j and the public key $pk_{\mathcal{J}}$ of $\mathcal{J}$. $\mathcal{S}$ inputs gpk, $pk_{\mathcal{J}}$, the secret watermarking key swk and the original content X. $\mathcal{S}$ obtains transaction information $info$ and stores it in the table entry Tab, where Tab is a table that stores information of all the transactions. $\mathcal{B}_i$ outputs watermarked content (reqresp, Y).

- **Arbitration.** When $\mathcal{S}$ is activated with (detect, Y), $\mathcal{S}$ runs $\mathsf{Detect}(swk, Y, \text{Tab})$ to obtain the table entry $info$ that corresponds to Y and sends ($info$) to $\mathcal{J}$. $\mathcal{J}$ runs $\mathsf{Check}(gpk, info, sk_{\mathcal{J}})$ to obtain a bit b and a deanonymization message M_{open}. If $b = 0$, $\mathcal{J}$ sends ($not\ guilty$) to $\mathcal{S}$ and outputs (detresp, $not\ guilty$). Otherwise $\mathcal{J}$ sends M_{open} to $\mathcal{D}$, which runs $\mathsf{Identify}(gpk, osk, M_{open}, reg)$ (reg is obtained from $\mathcal{R}$) and returns $\mathcal{B}_i$ and a proof τ that deanonymization was done correctly. $\mathcal{J}$ runs $\mathsf{VerifyId}(gpk, \mathcal{B}_i, \tau, upk_i, M_{open})$ to check the validity of the proof τ. If the output is $b = 0$, $\mathcal{J}$ sends ($\perp$) to $\mathcal{S}$ and outputs (detresp, $\perp$). Otherwise $\mathcal{J}$ sends ($\mathcal{B}_i$, $guilty$) to $\mathcal{S}$ and outputs (detresp, $\mathcal{B}_i$, $guilty$).

- $\mathsf{Request}(gpk, gsk_i, j, pk_{\mathcal{J}})$. Run $\mathsf{BKeygen}(1^k)$ to obtain a key pair $(sk_{\mathcal{B}}{}', pk_{\mathcal{B}}{}')$. Run $\mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}{}')$ to get an encryption C of $sk_{\mathcal{B}}{}'$. Pick a random string $W_{\mathcal{B}} \leftarrow \{0,1\}^{l_2}$ and, for $i = 1$ to l_2, run $c_i = \mathsf{BEnc}(pk_{\mathcal{B}}{}', W_{\mathcal{B}i})$ to encrypt bitwise $W_{\mathcal{B}}$. Set a message $m = (pk_{\mathcal{B}}{}', j, (c_i)_{i=1}^{l_2}, C)$ and run $\mathsf{GSsig}(gpk, gsk_i, m)$ to compute a signature s_m. (If m does not belong to the message space of the group signature scheme, use a collision-resistant hash function H to compute a hash $H(m)$ that belongs to the message space and sign $H(m)$.) Send (m, s_m) to $\mathcal{S}$. As the prover, engage with $\mathcal{S}$ in the following interactive zero-knowledge proofs of knowledge: a proof $\pi_1 = \mathsf{PK}\{(sk_{\mathcal{B}}{}') : (pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}') \leftarrow \mathsf{BKeygen}(1^k) \wedge C \leftarrow \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}{}')\}$ that $(pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}')$ are correctly setup and that C is an encryption of $sk_{\mathcal{B}}{}'$ under $pk_{\mathcal{J}}$; for $i = 1$ to l_2, a proof $\pi_{2i} = \mathsf{PK}\{(W_{\mathcal{B}i}) : c_i \leftarrow \mathsf{BEnc}(pk_{\mathcal{B}}{}', W_{\mathcal{B}i}) \wedge W_{\mathcal{B}i} \in \{0,1\}\}$ that each c_i encrypts a bit. Upon receiving ct, decrypt $Y = \mathsf{BDec}(sk_{\mathcal{B}}{}', ct)$ and output Y.

- $\mathsf{Response}(gpk, pk_{\mathcal{J}}, swk, X)$. Receive message (m, s_m). Parse m as $(pk_{\mathcal{B}}{}', j, (c_i)_{i=1}^{l_2}, C)$. Run $\mathsf{GSverify}(gpk, m, s_m)$ and abort if the output is 0. As the verifier, engage in the execution of the interactive proofs π_1 and, for

Setup	
1. $(gpk, isk, osk) \leftarrow \mathsf{GSgkg}(1^k)$	2. $\mathcal{B} : (upk_i, usk_i) \leftarrow \mathsf{GSukg}(1^k)$
3. $\mathcal{J} : (pk_{\mathcal{J}}, sk_{\mathcal{J}}) \leftarrow \mathsf{JKeygen}(1^t)$	4. $\mathcal{S} : swk \leftarrow \mathsf{WATsetup}$

Figure 3.10. The setup phase of the BSW protocol: 1) group key generation, 2) $\mathcal{B}$ key generation, 3) $\mathcal{J}$ key generation, 4) $\mathcal{S}$ sets up the watermarking scheme and obtains secret watermarking key

$i = 1$ to l_2, π_{2i}, and abort if any of them is not correct. Pick random $W_{\mathcal{S}} \leftarrow \{0,1\}^{l_2}$ and, for $i = 1$ to l_2, compute $\mathsf{BEnc}(pk_{\mathcal{B}}{}', W_{\mathcal{S}i} \oplus W_{\mathcal{B}i})$. Pick random unique $\phi \leftarrow \{0,1\}^{l_1}$ and, for $i = 1$ to l_1, encrypt $\mathsf{BEnc}(pk_{\mathcal{B}}{}', \phi_i)$. Set the watermark to be embedded as $W = \phi || (W_{\mathcal{S}} \oplus W_{\mathcal{B}})$, and let its bitwise encryption be $\mathsf{BEnc}(pk_{\mathcal{B}}{}', W)$. Perform the watermark embedding operation $\mathsf{WATemb}(swk, X, \mathsf{BEnc}(pk_{\mathcal{B}}{}', W))$ in the encrypted domain to obtain an encrypted watermarked content $ct = \mathsf{BEnc}(pk_{\mathcal{B}}{}', Y)$. Send (ct) and output transaction information $info = (\phi, W_{\mathcal{S}}, m, s_m)$.

- $\mathsf{Detect}(swk, Y, \mathrm{Tab})$. Execute the watermark detection algorithm WATdet (swk, Y) to obtain the watermark $W = \phi || x$, parse the table entry $(\phi, W_{\mathcal{S}}, m, s_m)$, compute $W_{\mathcal{B}} = W_{\mathcal{S}} \oplus x$ and output $info = (W_{\mathcal{B}}, m, s_m)$.
- $\mathsf{Check}(gpk, info, sk_{\mathcal{J}}, Y)$. Parse $info$ as $(W_{\mathcal{B}}, m, s_m)$ and m as $(pk_{\mathcal{B}}{}', j, (c_i)_{i=1}^{l_2}, C)$. Run $\mathsf{GSverify}(gpk, m, s_m)$ and abort if the output is 0. Decrypt $\mathsf{JDec}(sk_{\mathcal{J}}, C)$ to obtain $sk_{\mathcal{B}}{}'$. For $i = 1$ to l_2, decrypt $\mathsf{BDec}(sk_{\mathcal{B}}{}', c_i)$ to obtain $W_{\mathcal{B}}{}'_i$. Check whether $W_{\mathcal{B}}{}' = W_{\mathcal{B}}$. If it is the case, output $b = 1$ and $M_{open} = (m, s_m)$. Otherwise output $b = 0$ and $M_{open} = \perp$.
- $\mathsf{Identify}(gpk, osk, M_{open}, reg)$. Parse M_{open} as (m, s_m). Run $\mathsf{GSopen}(gpk, osk, reg, m, s_m)$ to obtain an identity $\mathcal{B}_i$ and a proof τ. Output $(\mathcal{B}_i, \tau)$.
- $\mathsf{VerifyId}(gpk, \mathcal{B}_i, \tau, upk_i, M_{open})$. Parse M_{open} as (m, s_m). Run $\mathsf{GSjudge}(gpk, \mathcal{B}_i, upk_i, m, s_m, \tau)$ to obtain a bit b. Output b.

3.7.3 Type III Setup Phase

As a preparation for the buyer-seller watermarking protocol, a setup phase is necessary to generate keys of $\mathcal{B}$ and $\mathcal{J}$, and group keys for the group signature scheme. The setup phase is presented in Figure 3.10. All messages are sent over a secure channel.

1. The Trusted Registration Authority executes the *group-key generation* algorithm GSgkg and generates a group public key gpk, an issuing key isk for

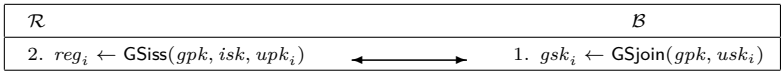


Figure 3.11. The registration protocol performed between the buyer $\mathcal{B}$ and the registration authority $\mathcal{R}$

the registration authority $\mathcal{R}$ to issue signature keys to group members, and an opening key *osk* for the deanonymization authority $\mathcal{D}$ to open signatures and retrieve the group member's identity.

2. The buyer $\mathcal{B}$ begins with the *user-key generation* algorithm GSukg to obtain a public and private key pair (upk_i, usk_i).
3. The judge $\mathcal{J}$ performs the *judge-key generation* algorithm JKg to obtain a public and private key pair (pk_J, sk_J).
4. The seller $\mathcal{S}$ executes the *watermarking setup* algorithm WATsetup to obtain secret watermarking key *swk* for the watermark embedding and detection. For instance, in a secure watermark embedding scheme based on dither modulation techniques [186, 250], the dithering value δ of can be considered as a secret parameter. In practice, one can use a different δ for each scalar feature, so that the secret key is actually a vector of dithering values. Besides, the set of marked features $\mathcal{M}$ can be considered as a secret parameter. Therefore, in such a watermarking scheme the secret watermarking key can be $K = (\delta, \mathcal{M})$.

3.7.4 Type III Registration Protocol

The registration protocol performed between the buyer $\mathcal{B}$ and the registration authority $\mathcal{R}$ is depicted in Figure 3.11.

1. To join the group, the buyer $\mathcal{B}$ executes the *group-join* algorithm GSjoin with the inputs (gpk, usk_i) and obtains a private signing key gsk_i.
2. The registration authority $\mathcal{R}$ executes the *group-issue* algorithm GSiss with the inputs (gpk, isk, upk_i) and stores the registration information reg_i in the registration table *reg*.

3.7.5 Type III Watermark Generation and Embedding Protocol

The watermark generation and embedding protocol can be executed multiple times for multi-transactions between the seller $\mathcal{S}$ and the buyer $\mathcal{B}$, as depicted

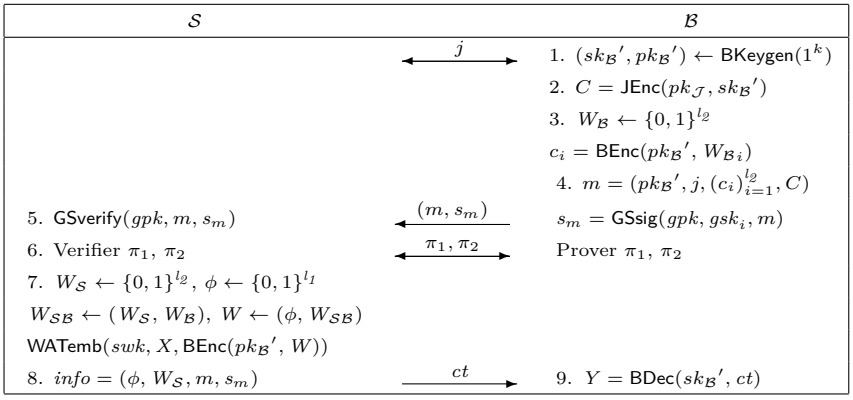


Figure 3.12. The watermark generation and embedding protocol of the Type III BSW protocol performed between the seller $\mathcal{S}$ and the buyer $\mathcal{B}$

in Figure 3.12. $\mathcal{S}$ and $\mathcal{B}$ first negotiate a purchase order j, which uniquely binds a particular transaction to the original digital content X.

1. The buyer $\mathcal{B}$ first runs $\mathsf{BKeygen}(1^k)$ to generate a one-time anonymous key pair $(sk_{\mathcal{B}}', pk_{\mathcal{B}}')$.

2. Next, $\mathcal{B}$ computes an encryption of $sk_{\mathcal{B}}'$ as $C = \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}')$, in order to let $\mathcal{J}$ retrieve $\mathcal{B}$'s private key in the arbitration phase.

3. $\mathcal{B}$ generates $\mathcal{B}$'s secret watermark as a l_2-bit random string $W_{\mathcal{B}} \leftarrow \{0,1\}^{l_2}$ and, for $i = 1$ to l_2, computes the bitwise encryption of $W_{\mathcal{B}}$ as $c_i = \mathsf{BEnc}(pk_{\mathcal{B}}', W_{\mathcal{B}i})$. The encrypted watermark is presented as $(c_i)_{i=1}^{l_2}$.

4. $\mathcal{B}$ sets a message $m = (pk_{\mathcal{B}}', j, (c_i)_{i=1}^{l_2}, C)$ and runs the *group-sign* algorithm $\mathsf{GSsig}(gpk, gsk_i, m)$ to obtain a signature s_m. $\mathcal{B}$ sends (m, s_m) to $\mathcal{S}$.

5. After receiving the message (m, s_m), $\mathcal{S}$ parses m as $(pk_{\mathcal{B}}', j, (c_i)_{i=1}^{l_2}, C)$, runs the *group-verify* algorithm $\mathsf{GSverify}(gpk, m, s_m)$, and aborts if the output is 0.

6. $\mathcal{B}$ as the prover and $\mathcal{S}$ as the verifier engage in the execution of the following two interactive zero-knowledge proofs of knowledge: a proof $\pi_1 = \mathsf{PK}\{(sk_{\mathcal{B}}') : (pk_{\mathcal{B}}', sk_{\mathcal{B}}') \leftarrow \mathsf{BKeygen}(1^k) \wedge C \leftarrow \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}')\}$ that $(pk_{\mathcal{B}}', sk_{\mathcal{B}}')$ are correctly setup and that C is an encryption of $sk_{\mathcal{B}}'$ under $pk_{\mathcal{J}}$; for $i = 1$ to l_2, a proof $\pi_{2i} = \mathsf{PK}\{(W_{\mathcal{B}i}) : c_i \leftarrow \mathsf{BEnc}(pk_{\mathcal{B}}', W_{\mathcal{B}i}) \wedge W_{\mathcal{B}i} \in \{0,1\}\}$ that each c_i correctly encrypts a bit. $\mathcal{S}$ aborts if any of the proofs is not correct.

7. $\mathcal{S}$ generates randomly $\mathcal{S}$'s secret watermark $W_{\mathcal{S}} \leftarrow \{0,1\}^{l_2}$ and an index $\phi \leftarrow \{0,1\}^{l_1}$ to locate the current transaction record in $\mathcal{S}$'s table Tab. Let $W_{\mathcal{SB}} = W_{\mathcal{S}} \oplus W_{\mathcal{B}}$, $W = W_{\mathcal{SB}} + \phi 2^n$. W consists of the l_2-bit $W_{\mathcal{SB}}$ and the l_1-bit ϕ. W can be decomposed into $l_1 + l_2$ binary numbers, with $W_i \in \{0,1\}$, satisfying $W = \sum_{i=0}^{l_1+l_2-1} W_i 2^i$, i.e., $W = \phi || (W_{\mathcal{S}} \oplus W_{\mathcal{B}})$. The watermark embedding can be considered as a function which takes the encrypted watermark bits $E_{pk_{\mathcal{B}}'}(W_i)$, the secret watermarking key swk, and the content X as input, and returns the encrypted watermarked content $E_{pk_{\mathcal{B}}'}(Y)$ as output, where $E_{pk_{\mathcal{B}}'}(\cdot)$ denotes $\mathsf{BEnc}(pk_{\mathcal{B}}', \cdot)$. The encrypted watermark can be computed in the encrypted domain as

$$E_{pk_{\mathcal{B}}'}(W) = \{E_{pk_{\mathcal{B}}'}(\phi_1), .., E_{pk_{\mathcal{B}}'}(\phi_{l_1})\} || \{E_{pk_{\mathcal{B}}'}(W_{\mathcal{SB}1}), .., E_{pk_{\mathcal{B}}'}(W_{\mathcal{SB}l_2})\} \tag{3.8}$$

where, for $i = 1$ to l_2

$$E_{pk_{\mathcal{B}}'}(W_{\mathcal{SB}i}) = E_{pk_{\mathcal{B}}'}(W_{\mathcal{S}i} \oplus W_{\mathcal{B}i}) = \begin{cases} E_{pk_{\mathcal{B}}'}(W_{\mathcal{B}i}) & W_{\mathcal{S}i} = 0 \\ E_{pk_{\mathcal{B}}'}(1) \cdot E_{pk_{\mathcal{B}}'}(W_{\mathcal{B}i})^{-1} & W_{\mathcal{S}i} = 1 \end{cases} \tag{3.9}$$

where $||$ denotes concatenation, and $\oplus$ denotes exclusive OR. $\mathcal{S}$ performs the watermark embedding $\mathsf{WATemb}(swk, X, \mathsf{BEnc}(pk_{\mathcal{B}}', W))$ in the encrypted domain to obtain an encrypted watermarked content $ct = \mathsf{BEnc}(pk_{\mathcal{B}}', Y)$. We have a generalized formula for watermark embedding in the encrypted domain

$$E_{pk_{\mathcal{B}}'}(Y) = E_{pk_{\mathcal{B}}'}[f(X)] E_{pk_{\mathcal{B}}'}(W)^{g(X)} \tag{3.10}$$

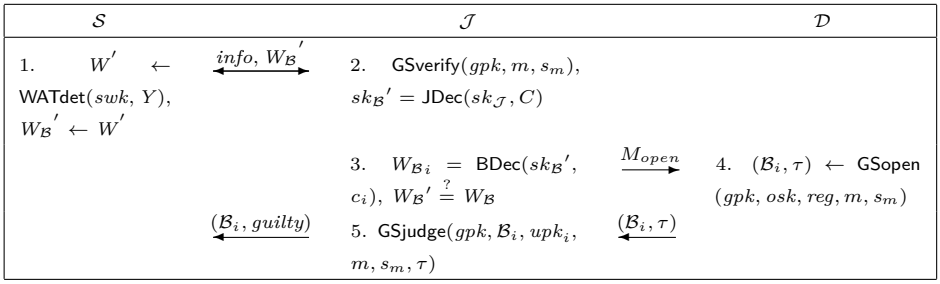
where $f(X)$ and $g(X)$ depend on the chosen watermarking technique, such as additive spread-spectrum schemes or QIM watermarking schemes.

8. $\mathcal{S}$ stores the transaction information $info = (\phi, W_{\mathcal{S}}, m, s_m)$ in $\mathcal{S}$'s transaction record table Tab, and delivers the encrypted watermarked content ct to $\mathcal{B}$.

9. Upon receiving ct, $\mathcal{B}$ decrypts $Y = \mathsf{BDec}(sk_{\mathcal{B}}', ct)$ and obtains the watermarked content Y.

3.7.6 Type III Identification and Arbitration Protocol

The identification and arbitration protocol, performed among the seller $\mathcal{S}$, the judge $\mathcal{J}$, and the deanonymization authority $\mathcal{D}$, is depicted in Figure 3.13.



Note: $m = (pk_{\mathcal{B}}{}', j, (c_i)_{i=1}^{l_2}, C)$, $s_m =$ GSsig(gpk, gsk_i, m), $info = (\phi, W_{\mathcal{S}}, m, s_m)$

Figure 3.13. The copyright violator identification and arbitration protocol of the Type III BSW protocol performed among the seller $\mathcal{S}$, the judge $\mathcal{J}$, and the deanonymization authority $\mathcal{D}$

1. Once a pirated copy Y of the original content X is found, $\mathcal{S}$ detects the watermark W' from Y and retrieves the most significant l_1 bits of it as an index ϕ' to search in $\mathcal{S}$'s table Tab, by choosing the ϕ from the table that is mostly correlated with ϕ'. Besides, $\mathcal{S}$ also recovers a watermark $W_{\mathcal{B}}{}'$ from $W' = \phi' || (W_{\mathcal{S}}{}' \oplus W_{\mathcal{B}}{}')$. $\mathcal{S}$ provides the collected information $info = (\phi, W_{\mathcal{S}}, m, s_m)$ and $W_{\mathcal{B}}{}'$ to the judge $\mathcal{J}$.

2. $\mathcal{J}$ receives *info* as $(\phi, W_{\mathcal{S}}, m, s_m)$ and m as $(pk_{\mathcal{B}}{}', j, (c_i)_{i=1}^{l_2}, C)$. $\mathcal{J}$ runs the *group-verify* algorithm GSverify(gpk, m, s_m), and aborts if the output is 0. If verified, $\mathcal{J}$ decrypts JDec$(sk_{\mathcal{J}}, C)$ to obtain $\mathcal{B}$'s private key $sk_{\mathcal{B}}{}'$.

3. For $i = 1$ to l_2, $\mathcal{J}$ performs bitwise decryption BDec$(sk_{\mathcal{B}}{}', c_i)$ to obtain bitwise $W_{\mathcal{B}i}$, and checks whether $W_{\mathcal{B}}{}' = W_{\mathcal{B}}$. If they match with a high correlation, $\mathcal{J}$ sends a court order $M_{open} = (m, s_m)$ to $\mathcal{D}$. Otherwise $\mathcal{J}$ sends (*not guilty*) to $\mathcal{S}$, and the protocol halts.

4. $\mathcal{D}$ executes the *group signature open* algorithm GSopen(gpk, osk, reg, m, s_m), where *reg* is obtained from $\mathcal{R}$, to obtain the identity $\mathcal{B}_i$ and a proof τ. Output $(\mathcal{B}_i, \tau)$.

5. $\mathcal{J}$ runs the *group signature judge* algorithm to GSjudge$(gpk, \mathcal{B}_i, upk_i, m, s_m, \tau)$ to check the validity of the proof τ. If τ is verified, $\mathcal{J}$ closes the case and sends $(\mathcal{B}_i, guilty)$ to $\mathcal{S}$. Otherwise, $\mathcal{J}$ returns $(\perp)$ to $\mathcal{S}$ and the protocol halts.

3.7.7 Zero Knowledge Proofs

The additive homomorphic cryptosystem used to encrypt the buyer's and the seller's watermark is Paillier's cryptosystem [230] which has been explained in Section 3.3.2.

Zero Knowledge Proof for Fair Encryption Of Private Keys π_1

In our protocol, $\mathcal{B}$ (as the prover) needs to convince $\mathcal{S}$ (as the verifier) that given the ciphertext $C = \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}{}')$ is an encryption of $\mathcal{B}$'s private key $sk_{\mathcal{B}}{}'$, such as the factorization of the modulus n, without revealing any secret information; and the trustworthy judge $\mathcal{J}$ is able to recover the buyer's private key, with the encryption C and $\mathcal{J}$'s private key $pk_{\mathcal{J}}$. Indeed, $\mathcal{B}$'s Paillier public key is $n = pq$ and g, and $\mathcal{B}$'s Paillier private key is $\lambda = lcm(p-1, q-1)$ which is equivalent to the factorization of the modulo n.

We employ the fair encryption of RSA keys by Poupard and Stern [248]. The encryption scheme of a private key and the proof of fairness work as follows:

Key Generation Let N be an RSA modulus $N = P \cdot Q$, where P and Q are primes, $gcd(N, \varphi(n)) = 1$, and G be an integer of order multiple of N modulo N^2. The third party CA's public key is (N, G), and the private key is $\lambda(N)$. The buyer's private key is $\lambda(n) = lcm(p-1, q-1)$, with the factoring components p and q such that $n = pq$, which is the modulus of the Paillier cryptosystem's between the buyer and the seller.

Encryption $\mathcal{P}$ (the buyer) computes $x = n - \varphi(n) = p + q - 1$, randomly chooses $u \in \mathbb{Z}_n^*$ and computes $\Gamma = G^x \cdot u^N \bmod N^2$.

Non-interactive proof The common inputs to $\mathcal{P}$ and $\mathcal{V}$ are randomly chosen integers $z_i \in Z_n^*$ for $i = 1..K$.

$\mathcal{P}$ randomly chooses $r_1, r_2 \in [0, A[$ and $v_1, v_2 \in Z_n^*$, and computes the commitment $t_1 = \left(G_{r_1} {v_1}^N \bmod N^2, (z_j^{r_1} \bmod n)_{j=1..K}\right)$,
$t_2 = \left(G_{r_2} {v_2}^N \bmod N^2, (z_j^{r_2} \bmod n)_{j=1..K}\right)$, and $e_1 = H\left(t_1, N, G, (z_j)_{j=1..K}, n\right)$, $e_2 = H\left(t_2, N, G, (z_j)_{j=1..K}, n\right)$. $\mathcal{P}$ computes $y_1 = r_1 + e_1(n - \varphi(n))$, $y_2 = r_2 + e_2(n - \varphi(n))$ and $s_1 = u^{e_1} \cdot v_1 \bmod N$, $s_2 = u^{e_2} \cdot v_2 \bmod N$. The non-interactive proof is a 6-tuple $(y_1, s_1, e_1, y_2, s_2, e_2)$.

$\mathcal{V}$ checks $0 \leq y_1 < A$ and $0 \leq y_2 < A$,
computes ${t_1}' = \left(G^{y_1} \cdot {y_1}^N / \Gamma^{e_1} \bmod N^2, (z_j^{y_1 - e_1 n} \bmod n)_{j=1..K}\right)$
and ${t_2}' = \left((G^{y_2} \cdot {y_2}^N / \Gamma^{e_2} \bmod N^2, (z_j^{y_2 - e_2 n} \bmod n)_{j=1..K}\right)$,
checks $e_1 = H\left({t_1}', N, G, (z_j)_{j=1..K}, n\right)$ and $e_2 = H\left({t_2}', N, G, (z_j)_{j=1..K}, n\right)$. $\mathcal{V}$ accepts if and only if this holds.

Zero Knowledge Proof for Bit Encryption π_2

The zero knowledge proof π_2 should be repeated l_2 times, where l_2 is the bit length of the buyer's watermark. The buyer (as the prover) needs to prove to the seller (as the verifier) that a given ciphertext C is an encryption of a bit without disclosing the bit value. Our proof protocol is based on the zero knowledge proof by Damgård and Jurik [108].

Since Paillier's encryption is $E(i) = g^i \cdot r^n \mod n^2$, and it can be seen a specialized form of the Damgård-Jurik cryptosystem. Given ciphertext c and two candidate plaintexts $w_1 = 1$ and $w_2 = 0$, $\mathcal{P}$ and $\mathcal{V}$ both compute $u_1 = cg^{-w_1} \mod n^2$ and $u_2 = cg^{-w_2} \mod n^2$. It is easy to see that the proof is equivalent to convincing $\mathcal{V}$ that either u_1 or u_2 is a n-th residue modulo n^2. We assume that $\mathcal{P}$ knows an n-th root u_1, and $\mathcal{M}$ is the honest-verifier simulator for the n-th residue modulo n^2 protocol.

The honest-verifier zero knowledge proof consists of two building blocks: 1) to prove a value is n-th residue modulo n^2, and 2) to prove a value is 1-out-of-2 n-th residue modulo n^2. To construct four-round perfect zero-knowledge proofs of knowledge based on honest-verifier zero knowledge proofs, we refer to the framework introduced by Cramer, Damgård, and MacKenzie [103].

Prove a value is n-th residue modulo n^2:
Common input: n, u
Private input for $\mathcal{P}$: v, such that $u = v^n \mod n^2$
1. $\mathcal{P}$ chooses at random $r \in \mathbb{Z}_n^*$, and sends $a = r^n \mod n^2$ to $\mathcal{V}$.
2. $\mathcal{V}$ chooses a challenge e, a random k-bit number, and sends e to $\mathcal{P}$.
3. $\mathcal{P}$ sends the response $z = rv^e \mod n^2$ to $\mathcal{V}$.
4. $\mathcal{V}$ checks that $z^n = au^e \mod n^2$, and accepts if and only if this holds. Otherwise, the protocol halts.

Prove a value is 1-out-of-2 n-th residue modulo n^2:
Common input: n, u_1, u_2
Private input for $\mathcal{P}$: v_1, such that $u_1 = {v_1}^n \mod n^2$
1. $\mathcal{P}$ chooses at random $r_1 \in \mathbb{Z}_n^*$, and then invokes $\mathcal{M}$ on input n, u_2 to get a conversation a_2, e_2, z_2. $\mathcal{P}$ sends $a_1 = {r_1}^n \mod n^2$, a_2 to $\mathcal{V}$.
2. $\mathcal{V}$ chooses a challenge d, a random t-bit number, and sends d to $\mathcal{P}$. Note that if k is the bit length of n, we can set $t = k/2$ and be assured that a cheating prover can made the verifier accept with probability $\leq 2^{-t}$.
3. $\mathcal{P}$ computes $e_1 = d - e_2 \mod n^2$ and $z_1 = r_1 v_{1e_1} \mod n^2$, and sends e_1, z_1, e_2, z_2 to $\mathcal{V}$.
4. $\mathcal{V}$ checks that $d = e_1 + e_2 \mod 2^t$, $z_1^n = a_1 {u_1}^{e_1} \mod n^2$ and $z_2^n = a_2 {u_2}^{e_2} \mod n^2$, and accepts if and only if this holds. Otherwise, the protocol halts.

3.8 Conclusion

Multimedia content distribution through the Internet has become a popular technology in today's digital world. While it provides many advantages to both customers and content providers, its main drawback is that it permits digital content to be illegally redistributed by dishonest users. One of the adopted strategies for deterring illegal redistribution of multimedia content is the buyer-seller watermarking BSW protocols based on digital watermarking technology and cryptography. A unique code is embedded into each copy of the distributed content, linking that content to a particular user or device receiving it. If an unauthorized content is found, the user who has redistributed the content can be traced by detecting the watermark. In additional, buyer-seller watermarking protocols provides copyright protection for the content providers, piracy tracing of the unauthorized content, and privacy protection for the customers. The protocol is anonymous when the identity of buyers is not revealed if they do not release pirated copies.

In this chapter, we defined the fundamental requirements of such watermarking protocols, proposed the constructions of three types of anonymous buyer-seller watermarking (BSW) protocols based on homomorphic encryption and group signatures. In contrast to earlier work, the proposed anonymous BSW protocols fulfill the desired security properties simultaneously.

Primarily, the Type I BSW protocol extends and improves its predecessors and ensures revocable anonymity for buyer and security for both buyer and seller. Homomorphic encryptions facilitate operations such as watermark embedding in the encrypted domain. Group signatures introduce piracy traceability, buyer's anonymity, and transactions unlinkability. Besides, anonymous communication channels enable both anonymous outgoing connections and anonymous hidden services. It also supports multiple transactions between buyers and sellers and doesn't require buyers to participate in the dispute resolution phase.

Moreover, the Type II BSW protocol improves the Type I BSW protocol for that watermark generation and embedding scheme, such that each of buyer and seller contributes a share of a secret composite watermark, however, none of them knows the exact watermark embedded in the original content. This improvement insured that the underlying watermarking scheme is not required to be linear or to tolerate the permutation of watermarks to permutation tolerant or linear watermarks, such that given a number of watermarked content Y_1 and Y_2, $\phi(Y)$ (ϕ is a permutation function) or $aY_1 + bY_2$ ($a, b \in \mathcal{R}$) is another valid watermarked content. In our protocol, we need functionality such that, given a vector of encrypted watermark bits $\mathcal{E}[w_i]$ and a content X, it is able to produce the encrypted and watermarked content $\mathcal{E}[Y]$. Every watermarking scheme that is invisible and robust to counter post image processing or malicious attacks that are possibly encountered later,

and supports the above functionality can be used with our scheme. Possible watermarking schemes include the spread-spectrum watermarking scheme, QIM, DC-QIM and RDM algorithms.

Finally, the Type III BSW protocol further improves the Type I and Type II BSW protocols by incorporating blind watermarking schemes, homomorphic encryptions, group signature schemes and several zero-knowledge proofs of knowledge as main cryptographic building blocks. Existing buyer-seller watermarking protocols are not provided with a formal analysis of their security properties. We have proposed a formal security definition for copyright protection protocols in the ideal-world/real-world paradigm. Furthermore, we have analyzed the security of an anonymous buyer-seller watermarking protocol and proven that it fulfills our definition. In particular, we have shown that the protocol is secure against any p.p.t. adversary when instantiated with a watermarking scheme, an encryption scheme, a group signature scheme and zero-knowledge proofs of knowledge that provide security against any p.p.t. adversary. Unlike the other building blocks, no watermarking scheme has been proven to offer this security level, and thus the actual security of the protocol against malicious buyers is lowered to the security offered by the watermarking scheme.

Chapter 4

Privacy-Friendly Architecture to Manage Distributed E-Health Information

4.1 Introduction

Recent years have drawn increasing attention from both industry and research communities towards the technological evolution of electronic health (e-Health) systems. The goals of these systems are threefold. A first goal is to provide ubiquitous access to lifelong clinical records of a patient to all relevant stakeholders, including the patient, anytime, anywhere, on any device. A second goal is to integrate and enrich the clinical, medical and operational knowledge to support lifelong health guidance of citizens within a community, region, and country. A third goal is to streamline the workflow into shared clinical and operational pathways in order to enable disease management and optimally support the clinical process. Combining these three goals facilitates inter-professional collaboration, while guaranteeing the privacy of the patient.

The major technical challenges facing e-Health services are facilitating efficiency, information retrieval and availability, and cross-context interoperability, without compromising the patient's privacy. The rapid aging of populations, combined with pressure on budgets for healthcare delivery, and technological advances are the driving forces behind these challenges. Hence, in the realm of e-Health, security and privacy issues have a deep impact. Privacy refers to protect private information, such as patient's ID and sensitive medical information, of certain

entities. Security techniques, such as access control mechanisms, are adopted in e-Health systems to ensure that only involved and properly authorized parties have access to sensitive data.

4.1.1 From Provider-Centric Towards User-Centric

Traditional e-Health solutions were mainly concerned with a limited view on the patient information, taking a provider-centric approach, and mostly limited to a single e-Health service provider. A paradigm shift is taking place in the e-Health domain, which is evolving from provider-centric towards user-centric healthcare.

The adoption of user-centric and privacy-friendly identity and information management systems can help to keep the number of parties who deal with a person's healthcare information as small as possible. For example, the circle of trusted parties should not be extended or broken by moving from a paper-world to an e-Health administration. A patient expects a trust relation with medics, however, as in the past with a doctor's secretary, the trust with a system administrator may not be the same as with medics.

In provider-centric identity and information management systems, patient's e-Health data is hosted and managed by a service provider using a central repository. This has various advantages from the service provider's point of view, such as being cost effective and easily scalable. The disadvantage is that by applying such an approach, the user loses control over the use of personal information. The user can regain this control with a user-centric identity and information management (IDM) system.

In user-centric identity and information management systems, the user is put in the center of interest and is given control over personal information. In particular, this means that the user can influence or even specify the policies that must be enforced when e-Health service providers wish to process his information. This has the obvious advantage of better protecting the privacy of each individual user. However, responsibility for storing and updating correct data then lies with the user.

4.1.2 Towards Interoperable and Privacy-Friendly E-Health Architecture

The Need for Interoperability

Interoperability and privacy protection have become important issues, especially as more and more healthcare service providers start collaborating online, using a

wide range of e-Health systems, certainly if they refer to the e-Health information stored in each other's systems.

Previous work mostly emphasizes the e-Health solutions from a single provider viewpoint, and reveals an unsatisfactory provision for the interoperability and privacy protection problem in cross-context IDM systems. This is why a generic interoperable and privacy preserving identity and information management framework is necessary, where each application domain may deploy a user-centric IDM system, allowing the collaboration and interoperation of a multitude of heterogeneous IDM systems.

In order to improve the quality of a patient's experience, one important requirement is the continuous and transparent availability of medical information, independent of the location where the information has been actually stored. Although a patient will typically visit different healthcare providers over time, and hence the medical information will be dispersed over several locations, the medical record of a patient should be available anytime and anywhere, in a location-independent way. To this aim, healthcare providers, such as hospitals, general practitioners, research laboratories, etc., should federate to share their e-Health information.

Privacy Concerns

Personal medical data is of sensitive nature, and therefore several laws and regulations mandate to protect the privacy of the patient, which is to be introduced in Section 4.2. In contrast to traditional e-Health systems where full trust is granted to service providers, a new trend for e-Health systems is to limit the trust in service providers to protect patient privacy. In fact, the aforementioned federation scenario presents two specific privacy threats, because it makes intensive use of identity information. One privacy threat is that, for instance, in order to retrieve all the necessary data relevant for the "treatment" of a patient, there must be a mechanism to cross-reference medical documents across several healthcare providers. That is, it should be possible to search and retrieve documents from several locations on the basis of the patient identity. Naturally, access to such documents is restricted by authorization rules, which, yet again, make an intensive use of identity information about both the healthcare professionals and the patients. Examples clarifying the role of identity in the authorization process are provided in Section 4.3.2.

From a functional perspective, the simplest solution would be use of global identifiers, such as national identification (ID) numbers, across different healthcare providers, or 'contexts' from this point on. However, this is not a feasible strategy for two reasons. First, healthcare providers would require maintaining control over the process of issuing identifiers. This is mainly due to legacy constraints,

as relevant legislation will be introduced in Section 4.2. Second, if medical data sources would use global identifiers, the risk of the privacy threat of linkability, such as massive data aggregation and profiling, would be much higher. An attacker that got to know the content of two medical databases could be able to correlate the data quite easily.

Instead of global identifiers, it is common practice for each healthcare provider to issue a unique identifier for an entity, such as a patient. When context-specific information is transferred from one context to another, the same information is expressed by means of different types or values. Typically, healthcare providers use different terms for the same entity, strictly relying on dictionaries may be very misleading. Besides, all information exchanged between healthcare providers in an e-Health system needs to be uniquely identified. Note that the above problem is a cross-context issue when global identifiers should not be shared directly among contexts, mainly due to legislation. Linking information from one context to another should not be straightforward, hence the need for a privacy-friendly but interoperable IDM system. Solutions to this problem will be explained in Section 4.4.

Another privacy threat is that, for instance, when staff from one healthcare provider (such as a generic hospital) retrieves a patient's e-Health information from another healthcare provider (such as a psychiatric hospital), to gain a complete overview of the patient's medical history. If we consider service providers untrustworthy, it is obviously not privacy friendly for the patient, because excessive personal e-Health information has been provided to the service provider. Therefore, a technical enforcement is needed in order to facilitate a privacy enhancing sharing of distributed e-Health information. This will be elaborated further in Section 4.5.

To accommodate these conflicting forces, namely the need of cross referencing and sharing documents and the avoidance of the aforementioned privacy threats, some solutions have been proposed that employ a mediating component, in which a mapping of context-specific identifiers and a conversion of context-specific information occurs when data is exchanged among different contexts. Local identifiers are used within each context and the mediator provides translation services from one context to another. However, if the mediator maintains the translation information on board, such as in the form of a lookup table, it becomes a likely target for attackers. An attacker could steal that information and use it to perform the correlation mentioned above. State-of-the-art solutions in the e-Health domain are vulnerable to such attack scenario.

4.1.3 Summary of Contributions

In this chapter, we introduce an interoperable and privacy friendly identity and information management framework that builds on previous work [18, 25, 24], and

on ongoing research [279, 39]. We instantiate this generic framework in the e-Health domain, while taking the specificities of the healthcare sector into account. These specificities include the ability to uniquely refer to a person across different medical domains such as a hospital, general practitioner, clinical research lab by means of unique identifiers or pseudonyms; the ability to allow a person to specify which actors are allowed to use his personal data by means of rule-based authorization; the ability to map information types and values used in one medical domain onto those that are semantically equivalent in another domain; and the ability to limit which information can be linked within and across medical domains.

In particular, we move towards this framework by defining a new model to manage identifiers and translate context-specific (or domain-specific) to ensure information interoperability, and introducing technical countermeasures to ensure data minimization and privacy enhancement in e-Health. Specifically, this chapter introduces a cryptographic algorithm to be used in issuing context-specific, hence local, identifiers. Local identifiers are derived from a unique global identifier in a reversible way. The algorithm is meant to be used by the identity providers located at each healthcare provider. Further, for cross-context interoperability, a state-less mediation service is presented. The mediation service leverages the reversibility property of local identifiers and does not maintain any cross-referencing information on board. Further, the entity that functions as the mediator is not fixed and may vary. Moreover, we ensure that only the minimum necessary patient's personal information is provided to the authorized parties, such that the data minimization principle is complied. This is accomplished by introducing a data anonymization mechanism to obfuscate the sensitive part according to the patient's privacy preference. This user-centric architecture is able to provide interoperability and privacy protection at the same time.

4.1.4 Publication Details

The work described in this chapter has been published as conference papers [118, 128], an international journal paper [120], and a book chapter [119].

4.1.5 Chapter Outline

In this chapter, legislation on privacy in e-Health and the use of national identification numbers, and the previous work are briefly reviewed in Section 4.2. The rest of the chapter outlines the e-Health user-centric architecture with an introduction to its functional components in Section 4.3.1, followed by a discussion of the relation between identity and authorization to access distributed personal e-Health information in Section 4.3.2, an explanation of the proposed interoperable and privacy enhancing identity and information management service,

and a definition of the reversible algorithm to issue and convert context-specific identifiers in Section 4.4. Next, how the proposed interoperable and privacy preserving information management framework can be integrated in an e-Health system is illustrated in Section 4.5, with an e-Health platform as a case study. In particular, the motivating scenario, the system model and players, the attack model and assumptions are defined. Services of each entity, the command flow for a service request, and the protocol with the proposed scenario are discussed accordingly in Section 4.5.3. Finally, a conclusion is provided in Section 4.6.

4.2 Background

4.2.1 Legislation and Standards

Medical data is of sensitive nature, and therefore several laws and regulations mandate to protect the privacy of the patient.

In 2006 the United States Department of Health and Human Service Health issued the Insurance Portability and Accountability Act (HIPAA) [16]. This is a regulation in healthcare to demand the protection of patients data shared from its original source of collection. Since 1995 the processing and movement of personal data is legally regulated by the EU with the Directive 95/46/EC [140]. A citizen's right of privacy is also recognized in the Article 8 [135] of the European Convention for the Protection of Human Rights and Fundamental Freedoms.

The debate surrounding the usage of single national identification numbers has longstanding historical roots. EU countries have sought to regulate their national number(s) in a variety of ways. Art. 8.7 of Directive 95/46/EC [140] provides that "Member States shall determine the conditions under which a national identification number or any other identifier of general application may be processed", indicating that governments should carefully consider how they allow national numbers to be used. Regardless of how national identification numbers are regulated in each respective State, they constitute "personal data" by nature that fall under the meaning of Directive 95/46. Art. 16 and 17 of the Directive that impose upon the controller a general confidentiality and security obligation, including the obligation for the controller to take all reasonable measures "to prevent all other unlawful forms of processing" (Art. 17). Regardless of the possible perception that this might lead to massive data aggregation and profiling by the government, on the value of which we would not make judgments, it is manifestly clear that the national number is not intended for use outside the governmental context.

4.2.2 Related Work

A number of user-centric identity management systems developed over the past years include Shibboleth [11, 263], Liberty Alliance [17, 27], U-Prove [252] and CardSpace [22], OpenID [226], Higgins Project [162], and Idemix [26, 85].

In the literature, there were some schemes proposed for e-Health utilizing a user-centric approach. Peyton et al. [236] used a simple ePrescription scenario to analyze the business and technical issues to be addressed in a Liberty Alliance-based federated identity management framework for e-Health. They look at the potential impact of privacy compliance on three existing components of the framework, namely, Discovery Service, Identity Mapping Service and Interaction Service. A fourth component Audit Service is proposed to address potential privacy breaches in Liberty Alliance. Au and Croll [57] proposed a new framework for a consumer-centric identity management for distributed e-Health. The healthcare consumer maintains a pool of pseudonym identifiers in their personal secure device for use in different healthcare services, perhaps in the form of a smart card. Without revealing consumer identity, health record data from different medical databases distributed in various points of clinical service can be collected and linked together on demand. In particular, pseudonym identifiers are cryptographically generated by a trustee, and the binding of an identifier to the identity key or another identifier is certified by a Key Binding Certificate issued by the trustee. Hence, security of the interactions among different entities in the architecture is guaranteed by certification and cryptographic technologies.

Some results have been published on privacy protection and secondary use issues of EHR (Electronic Health Record). Iacono et al. [164] discussed the importance of protecting the privacy of patient data kept in an EHR in cases where it leaves the control- and protection-sphere of the health care realm for secondary uses such as clinical or epidemiological research projects, health care research, assessment of treatment quality or economic assessments. The work focuses on multi-centric studies, where various data sources are linked together using Grid technologies. It introduces a pseudonymization system which enables multi-centric universal pseudonymization, meaning that a patient's identity will result in the same pseudonym, regardless of which participating study center the patient data is collected. Pommerening and Reng [244] addressed the issue of secondary uses of EHR, such as health economy and health care research, or disease specific clinical or epidemiological research. For these uses in general, the patient identity must be anonymized or pseudonymized. Their work describes possible model architectures, developed for medical research networks, but useful in broader contexts.

In Europe, there were several research projects on privacy and cross-border identity management. The concept of context-specific identifiers was introduced in the Modinis Study on Identity Management [18], which was an EU funded research project which focused interoperability aspects of identity management systems

used in the EU Member States. It aims was to build on expertise and initiatives in the EU Member States to progress towards a coherent approach in electronic identity management in eGovernment in the European Union. The study addresses interoperability issues in cross-context IDM in eGovernment, without ignoring differences in legal and cultural practices within the EU framework for data protection. GUIDE [15] was also an EU funded project aiming the creation of an architecture that will enable open and interoperable eGovernment electronic identity services in the EU. Its objective concerns interoperability across national systems and structures within broader transnational, policy, legislative, and socio-economic boundaries. The PRIME [28] project looked at the applicability issues of using the federated identity management system Idemix open source initiative and digital credentials in detail. The main contribution of this European research project is a broader understanding of the dependencies between the different components in such a system. These dependencies are reflected by both an identity management architecture and an integrated prototype. The PrimeLife [249] project builds upon and expands the foundation of the PRIME [28] project that has shown privacy technologies can enable citizens to execute their legal rights to control personal information in on-line transactions. PrimeLife resolves the core privacy and trust issues pertaining to two new privacy challenges: A first technical challenge is how to protect privacy in emerging Internet applications such as collaborative scenarios and virtual communities. A second challenge is how to maintain life-long privacy. To resolve these issues, PrimeLife aims substantially advance the state of the art in the areas of human computer interfaces, configurable policy languages, web service federations, infrastructures and privacy-enhancing cryptography. FIDIS [32] was a EU-sponsored Network of Excellence targeting various aspects of digital identity and privacy. FIDIS areas of interest includes new forms of ID cards, usage of identifiers in information systems, technologies used for citizen's identification and profiling. Research projects in Belgium, such as Identity Management for eGovernment [25], focus on the identity management aspects that are relevant in an heterogeneous eGovernment context and compare the different governments in Flanders, Brussels, and Wallonia that have to interoperate with the Federal services. The European TAS3 [279] Project aims to provide an integrated and context independent trusted services network that advances the current state of the art of isolated and context-dependent solutions, such as for developing service user-centric tools and programs related to e-Health. The goal is to provide a transparent framework in which process-based services can securely process and depend on personal information, regardless from the context in which this information was collected.

There are some governmental or industrial partners in Belgium active in the related fields of IDM or e-Health. The Crossroads Bank for Social Security [19] is active in the field of IDM of eGovernment in the social sector. This organization provides technical solutions to function as a mediator for cross-context communications among different sectors, and proposed an algorithm to issue one-way only context-

specific identifiers. Custodix [23] is a company active in the e-Health sector. Generally, Custodix is a Trusted Third Party that provides security solutions based on privacy enhancing techniques at international level. The services lay special emphasis on anonymization and pseudonymization.

4.3 Preliminaries

4.3.1 An E-Health Infrastructure

Classic community healthcare systems utilize an e-Portal functionality to provide a many-to-one connection between many GPs (general practitioners) and one hospital, based on propriety solutions. The *disadvantages* of this approach are twofold. First, it is impossible to interconnect different entities, such as hospitals. Second, one GP needs multiple portals to access patient data in different hospitals. As an improvement, a *centralized infrastructure* is enabled in community healthcare. Note that this does not need to imply that data physically resides in a central data store. The *advantage* of this approach is that users gain a consolidated overview on the clinical data of the patient, to which clinical research institutes and healthcare providers can interconnect.

To illustrate the concept, we present in this section an e-Health architecture developed in the E-Health Information Platforms (EHIP) project [24], which has been completed successfully in Flanders, Belgium, to create a clinical data sharing infrastructure among multiple healthcare providers. The e-Health information platform is designed with information, such as patient e-Health records, always available and accessible only to authorized actors, at the time and place it is required. Several key players in the healthcare section, including leading sector companies, several university research groups and large hospitals, have contributed to ensure that the research outcome is valid within a genuine context. In addition, a lifetime view is projected, which will be instrumental in guiding the transition in healthcare systems from provider-centric towards patient-centric.

The e-Health infrastructure aims to promote community healthcare and international standards on different fora. It provides a horizontal infrastructure for e-Health applications compatible with international standards of Cross-Enterprise Document Sharing [21] by Integrating the Healthcare Enterprise [33] and technologies such as Web 2.0 to be used in web-based portals [12, 245], which are interoperable within the Belgian e-Health digital platform Be-Health [13] infrastructure and hospital IT systems, with respect to security and privacy. On the other hand, it provides a vertical application-based prototype for hospitals and GPs to share a patient's Electronic Health Record (EHR), such as medical summary, clinical results and patient discharge letters.

Take in Figure 4.1 for the functional components of the e-Health platform. Based on a Service Oriented Architecture, where each subsystem exposes its functionality through a service interface, it utilizes a central document registry to contain the metadata of all available documents, and distributed document stores, where medical documents are stored in local repositories of the corresponding healthcare providers. This e-Health platform also contains a gateway to support healthcare providers with limited resources, such as small practices that cannot afford a repository. Further, the platform provides Internet-enabled access to the resources through a web portal, which facilitates actions such as accessing the platform after-hours. Documents in the platform share a common content model as Clinical Document Architecture [20], such that all parties, despite their heterogeneous internal systems, gain easy access. The architecture employs federated security, in which security is embedded in middleware. Federated policy enforcement at hospitals and GPs surgeries with a central policy management are deployed for access control, i.e., authentication and authorization, in compliance with the EU data protection directive. According to Cross Enterprise Document Sharing (XDS [21]), the EHR security is covered by the following *Integration Profiles*: the *Audit Trail and Node Authentication Profile* to provide audit trail and the *Cross Enterprise User Authentication Profile* to provide a federated identity management framework based on SAML (Security Assertion Markup Language) that enables Single Sign-On functionality across multiple enterprises [21]. A security enhanced software architecture with an analysis of potential security risks in the e-Health infrastructure has been proposed by Wuyts et al. [293].

4.3.2 Roles of Identity in Distributed E-health Network

It is well known that identification plays a key role in supporting authorization. From the study of typical authorization rules we realized that such role is even more fundamental in a distributed e-Health network. The e-Health platform described in Section 4.3.1 is a communication infrastructure that allows many healthcare providers to collaborate by sharing the medical information they produce. In collaboration with clinical partners, we have elicited and analyzed the low level policy rules used in a real hospital setting. Consequently, we have extracted the authorization rule types that are relevant in the federated case.

Roles have been adopted in the past as the cornerstone technique to manage permissions in e-Health, e.g., in the context of the UK National Health Service [219]. In fact, we observed that roles are less central than expected in deciding whether an access request to medical information should be granted or not. Rather, we discovered that existing relationships between patients and GPs, besides other context-dependent parameters, such as time and location, are of primary importance in the authorization process. Hence, establishing the identity of involved parties is often a primary pre-requisite to authorization. In the

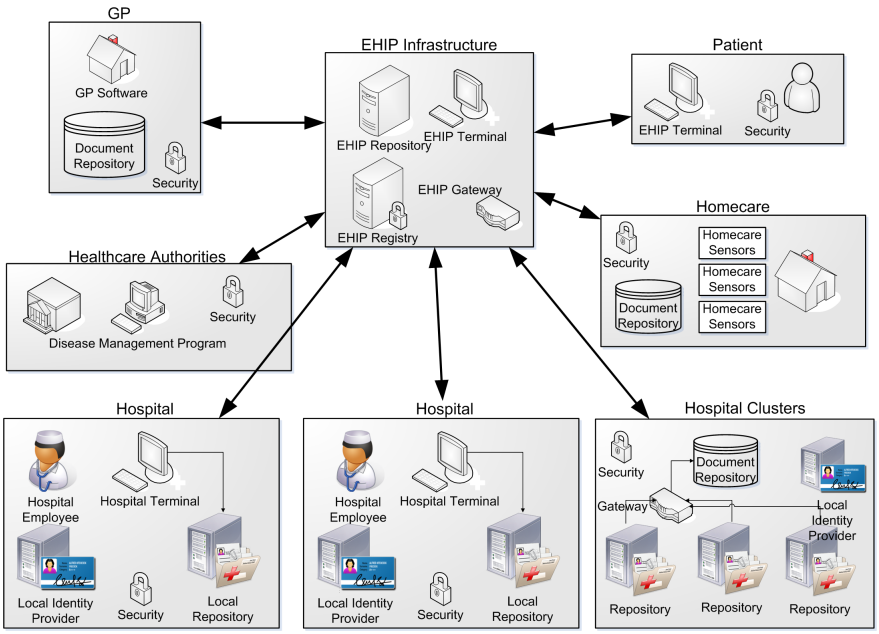


Figure 4.1. Bird-view of the e-Health infrastructure

remaining of this section we illustrate some typical policy rule types and highlight the identity-related information that is important for the decision process.

Rule-Based Authorization in Distributed E-Health Network

This section describes some generic authorization rules, each requiring the establishment of the identity of a specific patient in order to be enforced. Identity is typically used to verify the presence of a certain relationship between the patient and the GP requesting access to the patient data. Each rule type is described according to the same template: first we give a general description of the rule type, then we provide one example of a possible instantiation, and finally we provide a detailed explanation of the rule with particular focus on the role played by identity.

1) Patient-GP treatment relationship
Rule: GPs who treat a patient, either as supervisor or executing GP, are granted access to patient data related to that treatment.
Example: A screening center has access to the mammographic pictures of the

radiology center to perform a reading, because the screening center is implicitly treating the patient.

This policy provides an example of the treatment relationship, which is the relation between a patient and the GPs who are dealing with the patient during a treatment process. This relationship can be explicit or implicit. In an explicit relationship the treating GP is explicitly assigned, for instance by name, to the patient. Note that there is a clear relationship, as seen by both the GP and the patient. The implicit relationship is illustrated by the example, where the radiologist from the screening center is implicitly assigned to the patient by performing his function and can be considered as part of the treating process of the patient. Note that there is no direct relationship between the patient and the radiologist.

The policy will grant access to the patient data if a relationship exists, and will deny access if no relationship has been established. To decide whether or not a relationship exists, the identity of both the requester, such as a GP, and the patient must be established. Note that in a cross-context access request, identities are expressed in the 'vocabulary' of the requester, i.e., using identifiers that are local to the requester's context, which may not be meaningful to the authorization service of the context where the requested data belongs to.

2) Patient-department relationship
Rule: A GP is granted view access to the patient's data, if the patient resides or resided less than two weeks ago in a department to which the GP is assigned to.
Example: When a patient is transferred between hospitals, the GP of the hospital where the patient resided less than two weeks ago, can also access relevant data of the patient from the other hospital.

For this policy, the patient history has to be taken into account. The transfer of the patient between departments, or more in general, between healthcare institutions, needs to be tracked. The time the patient has spent in the hospital has to be considered as well. This policy is clearly related to the treatment relationship case. However, in this case, GPs no longer holding a current treatment relationship, can still access the patient's data.

3) GP-department relationship
Rule: A specific GP can view patient data that originated within one of the departments the GP is assigned to.
Example: A GP can remotely access data of the patient via a web portal if the data was created by the GP's department.

This policy is enforced by establishing the GP's affiliation. The example described above is rather narrow. This could be extended to data within the same discipline, spread over several healthcare institutions, instead of just within one department. Obviously, this rule requires that the patient-department relationship is verified, as in the previous case.

4) GP-patient relationship
Rule: A general practitioner (GP) retains the access to the medical reports concerning the patient as long as she remains registered as the patient's GP.
Example: A GP can always access medical reports of all of her patients.

A GP needs specialized rules, in contrast with other healthcare providers, because a GP does not belong to a healthcare institution. Therefore, the GP will not be granted access on the basis of a treatment relationship or because she belongs to a certain department. Rather, access decisions are only based on the long-lasting relationship with the patient.

5) Identity in obligations
Rule: A GP can overrule an access denial, provided that a detailed reason is specified. The system is obliged to log the identity, the reason, the access time, and the accessed resources.
Example: Before a surgical operation, an anesthetist does not automatically get access to the information about the allergies of a patient, because at that time the patient is not yet admitted, so the anesthetist is not a treating GP. An anesthetist can overrule the denial in order to better prepare for the operation. Overruled access is logged.

It is a strong requirement from the regulatory perspective to establish the identity of the GP who overruled the decision of the authorization service, and the identity of the patient for which such overruling took place. Therefore, policies exist describing what and how to log and they all require that the individual's identity is traced for auditing and possible legal reasons.

Identity and Authorization

An interesting result of this study is that role-based access control does not suffice in the distributed e-Health scenario. This section has identified several cases where verifying identity, rather than role-related credentials, is a pre-requisite to the enforcement of cross-context e-Health authorization rules. Further, in real world scenarios there are many, often complex, exceptions to the baseline rules described above, such as the following one: "no access to application X except for personnel of unit 500, for department PNE, LOG, PSY, unless they are assistants in training or if they have user-ID ABC or XYZ." This shows that identifiers play a key role in these cases.

In summary, the policies described above have illustrated that establishing identifiers is necessary to enforce authorization rules, which involve:

- current and historical treatment relationships: identities are used to evaluate the access rights of the GP on a need-to-know basis;

- visit history of the patient: identities are used to verify the relationship with a department, a discipline, and so on;
- long-lasting relationships: such as contractual relationships between patients and the GPs;
- exceptions: identities are directly referenced in the rules;
- auditing: identification is required by policy.

4.4 Proposed Architecture

In this section, we introduce an interoperable and privacy preserving architecture for identity and information management in e-Health. In general, there are two types of identifiers in the e-Health system: a global identifier of a patient, e.g., the national identification number, and context-specific identifiers. The context-specific identifiers in the system are used to locally identify a patient and the patient's medical record within a specific context, e.g., a healthcare provider.

4.4.1 Basic Concept

As mentioned in previous sections, all healthcare providers may have heterogeneous internal systems, and each healthcare provider typically issues its own unique *context-specific identifier* to patient as well as to the patient's medical record, that will be stored in the local repository of the corresponding healthcare provider. This means, the one patient will be issued different identifiers from different healthcare providers, similarly the patient's medical records stored in different healthcare providers will be assigned with different document identifiers. According to the legal restrictions explained in Section 4.2, it is not advised to share the patient's global identifier among contexts for privacy protection reasons.

We now attempt to expand the notion to multiple contexts interacting and communicating with each other. One complication that occurs is that administrations need to exchange information coming from different contexts. For example, one healthcare provider tries to query the medical record concerning a patient from another healthcare provider, such that context-specific information is exchanged from one context to another. Further, the personal information exchanged needs to be uniquely identified, but the same identifier should not be shared among contexts. Whenever information is exchanged between different contexts a mapping and conversion of identifiers is required. In order to exchange information between contexts, an identifier mapping and conversion is performed by a trusted party which is available for each context [18]. Since linkability of information from

one context to another is desirable but not yet feasible, a manageable system for information interoperability is required. Therefore, or goal is to provide a cross-context, and hence interoperable, and yet privacy-friendly system, compatible with all the internal systems employed by the entities in the e-Health platform, to translate and convert context-specific information and identifiers used and exchanged between the concerned entities.

Figure 4.2 depicts a structure to facilitate cross-context information sharing in e-Health. An administrative organization of a healthcare provider can be separated as front and back offices. The front office is connected with portals and local repositories. It directly interacts with its users, while the back office provides services for system support, such as identity management, authentication, authorization, information sharing, and auditing. The identity provider from the back office issues context-specific identifiers to its patients. Each healthcare provider is responsible for the issuance and use of context-specific identifiers within its context. Accordingly, one healthcare provider cannot prevent another healthcare provider from issuing context-specific identifiers for its patients within a particular context. When healthcare providers communicate, information can be exchanged through a mediating service provided by the e-Health platform. The mediating service, that is to say a trusted party available for each context, is responsible for mapping and converting context-specific information, such as identifiers, exchanged between the communicating parties. Note that here we not focus on how information is exchanged exactly, since it depends on semantic models and application or communication-specific scenarios. Instead, the contexts and entities involved in this communication are explored. In later sections, we explain how context-specific information can be converted and exchanged between contexts within a real-life scenario.

As shown in Figure 4.3, the abstract structure of the cross-context interoperable and privacy enhancing identity and information management system can be presented as an inter-connected solar system. In e-Health, node A, B, E, etc. denote back offices in healthcare providers. For instance, these can be the central server of a hospital, or a gateway connecting various portals of different departments of a hospital. When communicating cross-context between a healthcare service provider and a service requester, context-specific identifiers mapping and context-specific information conversion will be performed by a trustworthy agent called *mediator*, as denoted by node C, D, and G.

In the following sections, we introduce a reversible algorithm to issue and convert context-specific identifiers.

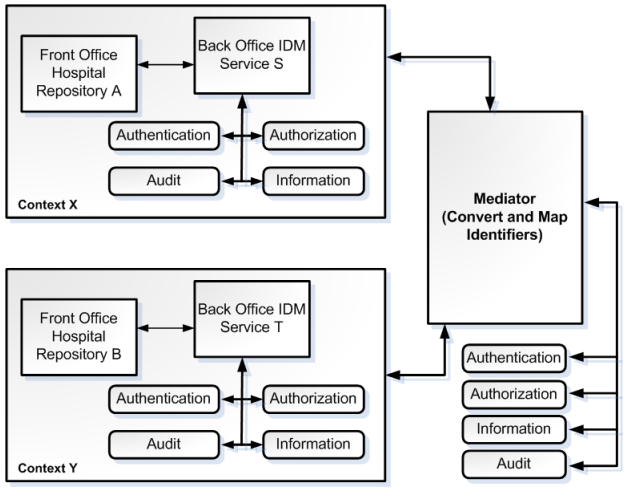


Figure 4.2. Cross-context information exchange in the e-Health infrastructure

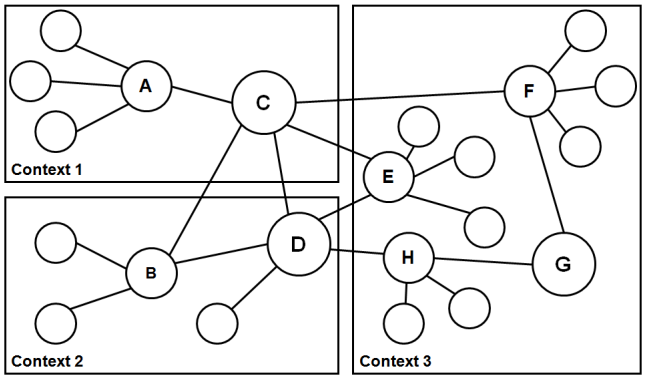


Figure 4.3. An abstract structure of an interoperable and privacy-friendly information management system with context-specific information conversion

4.4.2 Algorithms for Context-Specific Identifier Issuance

The algorithm to issue a context-specific identifier is run by the mediator in each context.

Definition 2. *`Anon` is a deterministic algorithm to issue a context-specific identifier. The algorithm's public inputs are the global identifier of an objective entity Gid of fixed-length, and a context-specific reference Ref of variable length.*

The private inputs are two symmetric secret keys K_e and K_h, for the pseudo-random function and the symmetric encryption function, respectively. The algorithm provides a fixed length context-specific identifier as:

$$\begin{aligned} Prefix &= MAC_{K_h}(Ref), \\ \textit{Anon}(Gid, Ref, K_e, K_h) &= E_{K_e}\big(MAC_{K_h}(Ref) \| Gid\big), \\ &= E_{K_e}(Prefix \| Gid). \end{aligned}$$

Take in Figure 4.4 for the construction of the context-specific identifier issuance algorithm. The context-specific reference of any length is the input to a pseudo-random function with a secret key, such as using a Message Authentication Code (MAC). For instance, with HMAC-SHA256, it results in a 256-bit message digest as a prefix. Then the prefix is concatenated to the global identifier, and encrypted using a symmetric encryption algorithm with a second secret key, such as AES in CBC mode [107]. The output is the context-specific identifier. Note that the secret keys (K_e, K_h) for encryption and pseudo-random function should be different.

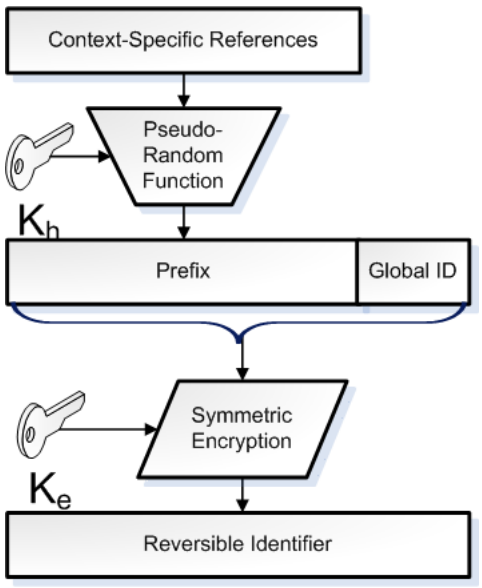


Figure 4.4. Algorithm to issue context-specific identifiers from the subject's global identifier

4.4.3 Algorithm for Context-Specific Identifier Conversion

The algorithm to extract the global identifier from a context-specific identifier is run by the mediator in each context.

Definition 3. *Deanon is a deterministic algorithm to extract a context-specific identifier from the objective entity's global identifier. The algorithm's public inputs are the context-specific identifier Aid, and the private input is the symmetric secret key K_e, for the symmetric decryption function. The algorithm provides a fixed-length global identifier as:*

$$\textit{Deanon}(Aid, K_e) = [D_{K_e}(Aid)]_{LSB_n}.$$

Note that the symbol $[X]_{LSB_n}$ refers to the extraction of the n least significant bits of X, as these correspond with the bits holding the global identifier.

Take in Figure 4.5 for the construction of the reverse process to convert a context-specific identifier from the global identifier. The context-specific identifier *Aid* is the input to a symmetric decryption algorithm, controlled by the secret key K_e. The result is the prefix concatenated with the global identifier *Gid*. The prefix is easily removed allowing the global identifier to be recovered.

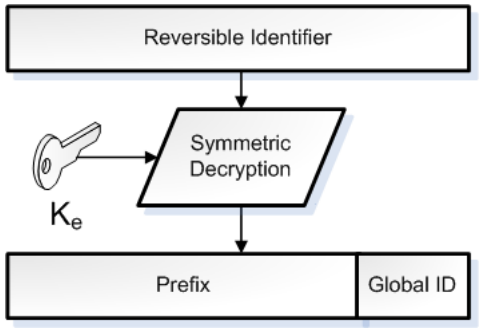


Figure 4.5. Algorithm to convert a context-specific identifier back to the subject's global identifier

4.5 Integration to the E-Health Infrastructure

In this section, we explain how context-specific identifiers can be converted and exchanged across-context in the e-Health platform, using a real-life scenario.

Consider the following scenario: suppose in the e-Health network, a medical staff in a generic hospital $\mathcal{H}_1$ intends to query a medical record of a patient $\mathcal{X}$ from a psychiatric hospital $\mathcal{H}_2$, through the e-Health *registry* $\mathcal{Y}$.

4.5.1 System model

With each entity denoting a context (i.e., an environment), we consider the scenario consists of the following entities:

1. *Healthcare provider* $\mathcal{H}_1$ is a generic hospital context, whose back office contains a file repository FR_1, a *patient ID provider* PIP_1, a *document ID provider* DIP_1, and a *document anonymizer* DA_1.
2. *Healthcare provider* $\mathcal{H}_2$ is a psychiatric hospital context, whose back office contains a file repository FR_2, a *patient ID provider* PIP_2, a *document ID provider* DIP_2, and a *document anonymizer* DA_2.
3. *Patient* $\mathcal{X}$ is a subject that requests healthcare services from a healthcare provider and the holder of an electronic health record (EHR) $Doc_{\mathcal{X}j}$. Let *Gid* denote $\mathcal{X}$'s global identifier, e.g., the patient's national number. Let $Pid_{\mathcal{X}j}$ denote $\mathcal{X}$'s context-specific identifier in a context j, namely a pseudonym assigned by the patient ID provider PIP_j of the healthcare provider $\mathcal{H}_j$. $Did_{\mathcal{X}j}$ denotes $\mathcal{X}$'s electronic health record $Doc_{\mathcal{X}j}$'s document ID, namely a pseudonym assigned by the document ID provider DIP_j to be used in the healthcare provider context $\mathcal{H}_j$.
4. *E-Health Registry* $\mathcal{Y}$ is a central registry that maintains a link between a patient's global ID and the locations of each healthcare provider that stores the patient's medical record.

4.5.2 Attack model and assumptions

In the aforementioned scenario, trust to the healthcare provider $\mathcal{H}_1$ and the e-Health network service provider (such as the system administrative staff) are limited. Therefore, an attacker to the system can be located inside any healthcare provider or from the system support staff. That is Eve, as the attacker, can be either internal to the e-Health network, or external. An internal attacker can either be an authorized or unauthorized recipient of the e-Health system services. We assume all the external attackers outside the e-Health network are unauthenticated and unauthorized entities to the system.

Eve's objective is to obtain any private information from a particular patient. In particular, Eve tries to obtain either the patient's global ID or the patient's

sensitive medical information from the psychiatric hospital $\mathcal{H}_2$. In order to do so, Eve has several options: first, she tries to request the patient's global identifier from the identity providers of each healthcare provider or the central registry. Then, Eve tries to request the sensitive medical data directly from the hospital $\mathcal{H}_2$. Afterwards, Eve tries to steal the secret keys of any identity provider or the document anonymizer in the system, in order to access the sensitive data. In addition, Eve tries to break into the system. Finally, she could try to eavesdrop the communication channel to obtain the desired content.

The following assumptions hold for the proposed system. All entities that have been authenticated and authorized by the system are assumed trustworthy. The system is not protected against malicious entities that are able to authenticate themselves and who are authorized to use the system's services. We assume that all security-enhancing functionalities employed in the system are robust and well-deployed. All secret keys of the entities in the system are stored physically secure. The communication takes place through a secure (i.e., privacy and authenticated) communication channel.

4.5.3 Proposed Approach

Services provided by each entity

Consider in a hospital $\mathcal{H}_j$, let K_{Pj} denote the secret key of the *patient ID provider* PIP_j, and let K_{Dj} denote the secret key of the *document ID provider* DIP_j. The ID providers PIP_j and DIP_j are able to provide the `IDIssue` and the `IDConvert` services. Let K_{Docj} denote the secret key of the *document anonymizer* DA_j. DA_j is able to provide the `DocAnon` and the `DocDeanon` services. Both $\mathcal{H}_j$'s file repository FR_j and the e-Health central registry $\mathcal{Y}$ can provide the `Query` service. Each kind of service is described as follows:

1. `IDIssue:` is a service to issue context-specific identifiers. Let $K = \{K_e, K_h\}$, then,

$$Aid = \texttt{IDIssue}(Gid, Ref, K) = \texttt{Anon}(Gid, Ref, K).$$

2. `IDConvert:` is a service to convert context-specific identifiers back to the global ID.

$$Gid = \texttt{IDConvert}(Aid, K) = \texttt{Deanon}(Aid, K_e).$$

3. `DocAnon:` is a service to pseudonymize part of a document *Doc* by encryption, which contains sensitive medical information. In other words, this is a data

anonymization algorithm to obfuscate the sensitive part, according to the patient's privacy preference or privacy policies.

$$ADoc = \texttt{DocAnon}(Doc, K) = E_{K_e}(Doc).$$

4. `DocDeanon:` is an algorithm to convert a pseudonymized document back to the non-pseudonymized version by decryption.

$$Doc = \texttt{DocDeanon}(ADoc, K) = D_{K_e}(ADoc).$$

5. `Query:` is a database query service with the input of some attributes and the output of some other attributes from the database.

Proposed approach to request a service

As depicted in Figure 4.6, each time before a service is delivered to a service requester from a service provider, first of all, the service requester needs to be authenticated and authorized.

1. A service requester sends a service request to a service provider.
2. The service provider forwards the request to its security facade to check the requester.
3. The security facade checks the requester's authenticity and authorization.
4. If the checks are passed, the security facade informs the service provider to deliver the required service. Otherwise, the service delivery is denied.
5. The service provider delivers the service to the service requester.

Protocol of the proposed scenario

Figure 4.7 presents the protocol of the scenario that a *Hospital* $\mathcal{H}_1$ queries a medical record of a patient $\mathcal{X}$ from *Hospital* $\mathcal{H}_2$ through the *registry* $\mathcal{Y}$ in the e-Health network. Note that $\mathcal{Y}$ serves as the mediator for the cross-context communication between $\mathcal{H}_1$ and $\mathcal{H}_2$, while interacting with the ID providers of the two contexts and performing context-specific identifier issuance and conversion.

Take in Figure 4.8. Information is transferred cross-context in the following steps:

1. $\mathcal{H}_1$ queries $\mathcal{Y}$ with the context-specific patient ID $Pid_{\mathcal{X}1}$ of a patient $\mathcal{X}$.

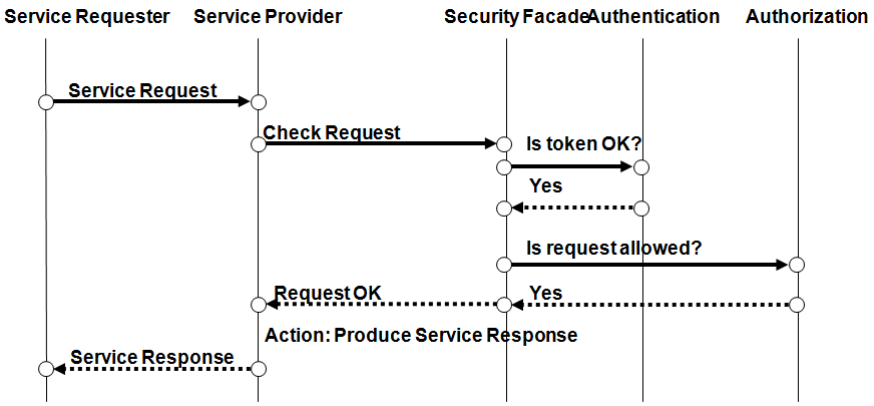


Figure 4.6. Check service request commands flow

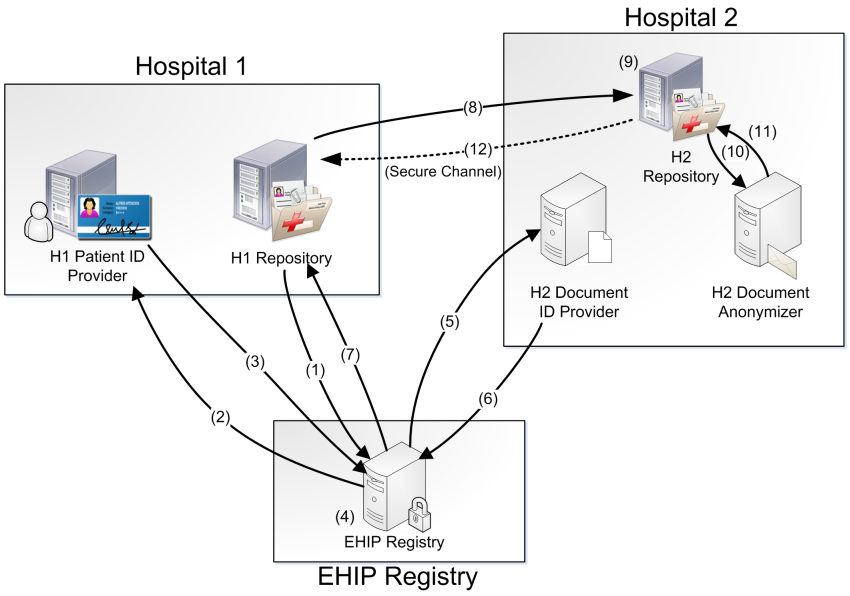


Figure 4.7. The protocol of the scenario of cross-context sharing of an electronics health document in the e-Health infrastructure

2. $\mathcal{Y}$ requests the `IDConvert` service with $Pid_{\mathcal{X}1}$ from $\mathcal{H}_1$'s patient ID provider PIP_1, in order to receive the global ID $Gid_{\mathcal{X}}$ of $\mathcal{X}$.

3. After the authentication and authorization check of $\mathcal{Y}$ by $\mathcal{H}_1$, PIP_1 runs $\texttt{IDConvert}(Pid_{\mathcal{X}1}, K_{P1})$ and delivers the result $Gid_{\mathcal{X}}$ to the registry $\mathcal{Y}$.
4. $\mathcal{Y}$ queries its database *Reg* to retrieve the corresponding location of the hospital where $\mathcal{X}$'s medical record is stored $Loc(\mathcal{H}_2) \leftarrow \texttt{Query}(Gid_{\mathcal{X}}, Reg)$, assuming that the document is at $\mathcal{H}_2$.
5. $\mathcal{Y}$ requests the `IDIssue` service from $\mathcal{H}_2$'s document ID provider DIP_2 with $Gid_{\mathcal{X}}$.
6. After the authentication and authorization check of $\mathcal{Y}$, DIP_2 provides $\mathcal{Y}$ the context-specific document ID of $\mathcal{X}$'s medical record stored in $\mathcal{H}_2$ by performing $Did_{\mathcal{X}2} \leftarrow \texttt{IDIssue}(Gid_{\mathcal{X}}, K_{D2})$.
7. $\mathcal{Y}$ sends $Did_{\mathcal{X}2}$ and $\mathcal{H}_2$'s location information $Loc(\mathcal{H}_2)$ to $\mathcal{H}_1$.
8. $\mathcal{H}_1$ sends a query to $\mathcal{H}_2$ with $Did_{\mathcal{X}2}$.
9. $\mathcal{H}_2$ queries its file repository FR_2's database $TabH_2$ and retrieves $\mathcal{X}$'s medical record Doc_{A2}, $Doc_{\mathcal{X}2} \leftarrow \texttt{Query}(Did_{\mathcal{X}2}, TabH_2)$.
10. $\mathcal{H}_2$ requests the `DocAnon` service from its document anonymizer DA_2 for an anonymized version of $Doc_{\mathcal{X}2}$.
11. After the authentication and authorization check, DA_2 performs the document pseudonymization $AnonDoc_{\mathcal{X}2} \leftarrow \texttt{DocAnon}(Doc_{\mathcal{X}2}, K_{Doc2})$, and obtains a pseudonymized health document $ADoc_{\mathcal{X}2}$.
12. $\mathcal{H}_2$ delivers the pseudonymized health document $ADoc_{\mathcal{X}2}$ to $\mathcal{H}_1$ through a secure channel.

4.5.4 Security Discussion

In this section, we analyze security of the proposed solution, in correspondence to the attack models. We will show, under the predefined assumptions, those potential threats described in Section 4.5.2 cannot be performed successfully in our proposed protocol setting. First of all, it is important to recall two assumptions. Primarily, all entities that are authenticated and authorized to obtain a certain service from a service provider are trustworthy, and vice versa. Additionally, any entity that fails to pass a service provider's authentication or authorization check should not obtain the corresponding service.

As explained above, it is obvious that the proposed architecture cannot be protected against a malicious entity who has been checked as an authenticated and authorized party. Considering the attacker Eve as an unauthenticated or

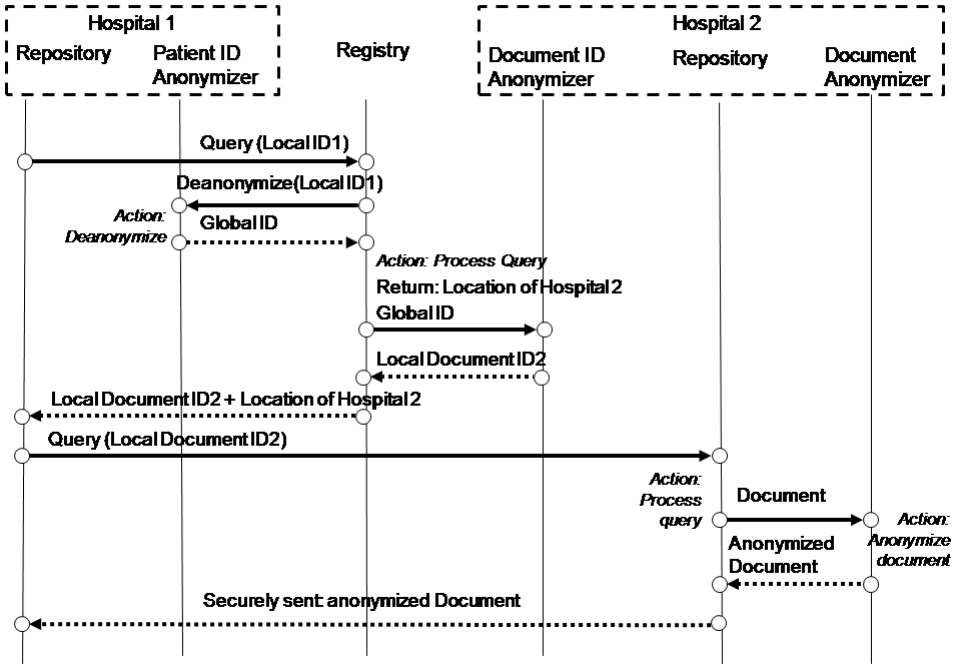


Figure 4.8. The command flow in the scenario of the cross-context document sharing in the e-Health infrastructure

unauthorized entity, we examine the five misuse cases that Eve may perform. In the first and the second misuse cases, Eve tries to request the patient's global identifier with the `IDConvert` service from any of the healthcare provider's identity providers or from the central registry. Or, she tries to request the sensitive medical data directly from the hospital $\mathcal{H}_2$. However, neither case is possible, because in order to receive the service, Eve has to pass the authentication and authorization check by the service providers. The third attack Eve may perform is to attempt to steal the secret keys of any identity providers or the document anonymizer in the system, such that she could perform the `IDConvert` service or the `DocDeanon` service to obtain the desired information. This is infeasible for Eve, since we assume all secret keys of the entities in the system are stored securely. In the next misuse case, Eve may try to hack the system and breakdown the security. This option is not feasible either, because all the security-enhancing functionalities employed in the system are assumed to be robust and well-deployed in the proposed system. If Eve fails to perform all the above attacks, Eve can still try to eavesdrop on

the communication content. However, presumably, the communication is taking place through a secure communication channel with client-side and server-side authentication.

Due to space limitations, a formal proof of security of the proposed scheme will not be provided in this chapter. As future work, for instance, one could prove that the security of the proposed system depends on the security of the secret key, the security of the communication channel, and the security of the underlying system security infrastructure, such as the security of the authentication and authorization mechanisms.

4.6 Conclusions

In this chapter, we presented an interoperable and privacy friendly architecture to manage the sharing of distributed personal e-health information. Previous e-Health solution mainly have a limited view on patient information, where a provider-centric approach usually was restricted to a single healthcare provider. Interoperability and privacy protection in the healthcare systems become more problematic in an e-Health infrastructure when more actors collaborate, such as hospitals, GPs, clinical research labs, pharmacists and so on. In particular, the protection of a patient's identity information and the patient's sensitive health data against unauthorized entities are the two main privacy concerns. However, the issues of interoperability and privacy protection have not been addressed in the state of the art. In this chapter, we proposed the technical enforcements to countermeasure these threats.

In the proposed setting, each patient is issued with a different and unique context-specific identifier by each healthcare provider. In communications across different healthcare providers (or contexts), the same information can be expressed by means of different types or values. Since patient's identity is not shared between contexts directly, linkability from one context to another should not be straightforward. To accommodate these conflicting forces, namely on one hand to protect patient's ID and personal e-Health information against unauthorized parties (even those within healthcare providers), and on the other hand to facilitate the possibility to follow-up a patient's medical treatment history by authorized parties, we introduce the concept of the mediating service to map and convert context-specific identifiers or information, when data is exchanged among different authorized healthcare providers. Further, we ensure that only the minimum necessary patient's personal information is provided to the authorized parties, such that the data minimization principle is complied. This is accomplished by introducing a data anonymization mechanism to obfuscate the sensitive part according to the patient's privacy preference. This user-centric approach is able to satisfy both interoperability and privacy protection.

At the algorithm level, we proposed an algorithm for issuing and converting context-specific identifiers, based on cryptographic techniques. As an illustration of the concept, we presented the e-Health architecture with a real-life use case scenario to explain how the proposed architecture can be integrated in the e-Health platform.

Chapter 5

Personal Rights Management for Individual Privacy Enforcement

5.1 Introduction

Over the last years, privacy protection has become a major issue, and both the European Union and the US are investing significantly into research on this area. However, almost all of the current work assumes an asymmetric model; the privacy violator is a corporate or governmental institution (or at least an employee thereof), while the victim is a normal citizen. Correspondingly, the main research areas cover issues such as identity management, policy enforcement, and anonymous communication. In the last years, however, a new privacy threat has emerged that cannot be addressed by such means. Due to improvements as well as the growing distribution of various handhold devices, an increasing number of people are equipped with miniature cameras (in their mobile phones) and voice recorders (in their music players).

Until the 1990s, public distribution of images could only happen in the press, either in print or in electronic broadcast media. To challenge the unauthorized distribution of an individual's image, a media company could be identified and contacted. Furthermore, the media company usually would know who the photographer was.

With the advent of the Internet as a public communication platform, fast and global distribution of images in public with web pages became common means.

Scanned photos then were available from an unknown number of private web pages. The availability of digital cameras reduced the cost and shortened the time it took to put images online. However, due to the physical dimension pointing a digital camera at a person can still be noticed in many situations.

In recent years, camera-phones were introduced. The build-in camera lens on a mobile phone can hardly be recognized, which brings the possibility that anyone who holds a camera-phone in an individual's surroundings could be taking a photo of the individual without being noticed. The individual won't be able to see a camera while being photographed or filmed, and won't know whether his images are put on the web or not.

With massive numbers of camera-phones out in the public, photos can be taken at any place. News stories about offenders being caught while shooting photos under women's dresses in public are available from the United States, Japan, Great Britain, Malaysia or even Saudi Arabia. Web sites like voyeurweb.com have been around longer than digital camera phones even exist to commercially distribute the content. While this intrusive and offensive use of cameras is regarded as illegal in many places in the world, other uses seem to create benefits for society – other news stories tell of offenders being identified thanks to camera-phone photos taken by bystanders of a crime. Considering the favorable uses of camera-phones in public, a solution that does detect, but not prevent from taking photos in public places may seem appropriate.

It has already shown to be a significant problem. At some beaches and in various companies, camera-phones are completely banned, and a number of countries have significantly increased the penalty for illegally taken pictures. Unfortunately, these countermeasures are by far not sufficient, as a growing number of web sites boasting such pictures demonstrates. As it is impossible and unwanted to enforce a broad ban on camera-phones, and technical measures such as a simulated shutter noise when a picture is taken appear to be insufficient, we propose a novel way to complement such measures.

This chapter deals with the challenge of protecting one's private data, such as image, and privacy issues attached to it. With respect to new mobile technologies and distribution channels, we sketch a privacy threat posed by millions of privately owned cameras in mobile phones.

Instead of preventing the picture from being taken, or call attention on the photographer when he takes the picture, we attack the distribution channel: if an inappropriate picture of an individual is taken and published, the victim has a fair chance of being the first one to actually find this picture, which enables her to request the pictures removal or invoke legal actions before significant privacy violation is done. The authors are aware that in extreme cases it will be impossible to remove a picture from the Internet by legal means. However, we expect that most of the privacy violations we address occur in a context where the publisher

could be convinced to remove the offending material without legal escalation. To achieve this, we propose that each picture receives an identity, which is contained in the picture and broadcasted to the victim that is photographed. Although this approach may be insufficient against a highly dedicated attacker, it can help to prevent privacy violations from becoming a mass phenomenon, without inhibiting the use of camera-phones, motivating users to manipulate their devices, or significantly increasing the costs of the devices.

5.1.1 Examples of legal context

Because of the fast growth of Internet new technologies as well as the incompatible policies between the different countries, in this context privacy issues are complex. From a technical perspective, Directive 95/46/EC of the European Parliament and the Council of 24 October 1995 [140] describes the protection of individuals regarding the processing and free movement of their personal data.

The right to privacy in the *EU* is defined as a human right under Article 8 of the 1950 European Convention of Human Rights and Fundamental Freedoms (ECHR). The implicit principles and constructs of The Directive define the enforcement and the representation of data protection. The terms privacy and data protection are often used interchangeably, though they are not necessarily equivalent. The Directive applies to all sectors of public life, with some exceptions. It specifies the data protection rights afforded to "data subjects", plus the requirements and responsibilities obligated for "data controllers" and by association "data processors" [140].

Several countries enacted laws against unauthorized taking of photos with individuals. More countries are debating legislation that is intended to ban camera-phones or their use. Some examples are given below.

In *Germany*, a copyright law ("Kunsturhebergesetz") protects one's own image against unauthorized publication since Bismarks's times. Photos can legally be taken without authorization, but their distribution without authorization, even to small audiences, is illegal. Exceptions are photos taken in public places at events where (press) photography usually happens. Also, individuals of "public interest" (e.g. politicians, actors, celebrities) can be photographed and published with limited restriction (see [133]).

In *Australia*, under the Commonwealth Crimes Act 1914 - Part VIIB, Section 85ZE it is an offence for "a person to knowingly or recklessly use a telecommunications service supplied by a carrier in such a way as would be regarded by reasonable persons being, in all the circumstances, offensive". In addition, following the widespread introduction of the Internet, state laws were changed to address this issue. For example the Crimes Act in Victoria was amended in 1995 to include the

offence of "Stalking". This includes telephoning and sending electronic messages with the intention of causing physical or mental harm.

While many countries do have legislation about camera based privacy invasions and the distribution of photos without consent of the photographed individuals, the question of the enforcement remains.

5.1.2 Current Solutions

The problem of secret photography has been recognized by most of the involved parties, including the manufacturers, politics and private citizens. Some actions have been taken, though with limited effect.

One solution is to fortify the privacy right on personal pictures and increased the punishment for the publication of such by *tougher laws*. However, this right may be hard to enforce. The photographed individual may never find out about the publication nor could do anything about it. Even though an offender may be caught on the scene, the phone could have already digitally transmitted the photo away. Even with laws enacted, the only choice of an individual would be to arrest the offender instead of waiting for the police to show up. This is not a setting that helps all members of a society with their privacy rights.

The second approach is to *ban the use of camera-phones* in places, such as public swimming pools, gyms and saunas. Though banning camera-phones could be the first choice in some places, this approach is only suitable for controlled areas with a high risk of secret photographing, such as companies or confidential institutes to counter espionage. The approach has also lead to the situation that even some mobile phone producers banned their own devices from their premises, e.g. Samsung and Motorola.

A more common sense solution is to add a sufficient loud *shutter-noise* such that whenever a picture is taken, it can be noticed by the environment. However, the feature is often poorly implemented. For example, if a mobile phone is switched into silent mode, the shutter noise is also turned off. Besides, given the noise pollution created by mobile phones anyhow, adding shutter noise can add to the annoyance of the technology. More, it violates the privacy of the photographer, as people around immediately learns about who being present with a camera. The approach is mostly ineffective, because the noise can be hard to hear due to general background noise or the environment, e.g. in a Discotheque, and it usually does not help the victim.

Given the difficulty to prevent pictures from being taken without dramatically infringing the rights of harmless photographers, our approach targets the distribution channel rather than the creation of the picture, i.e. pictures can be taken without restriction. However, the individual is made aware that some picture

has been taken. As soon as the picture appear on the Internet, she has a realistic chance to locate it at an early point in time, when it is still possible to inhibit the distribution by legal means. As an added value, outside of protecting the victim's privacy, this technology can also be used to distribute pictures to interested parties.

Another solution is to *enforce safe zones by broadcast*. Several businesses have developed a so-called safe haven technology which is intended to create zones where a broadcast unit tells camera-phones that photographing is forbidden there [267]. It enables digital cameras within a variety of electronic devices to be disabled including camera phones, camera PDA's, digital cameras and multipurpose MP3 players. HP is developing a privacy technology that can jam still and video cameras and blur faces of people who don't want to have their picture taken [255]. While this approach empowers property owners to define non-photographing zones, it also restricts a user's freedom of taking pictures with consent in the area. Another problem is that here is a need to implement the receiver technology into all manufacturers' handsets for an effect. Furthermore, to protect individual rights, one needs a portable unit. This only could guarantee personal rights independent from one's property protection policy.

5.1.3 Conflict of Interest

In order to protect the privacy rights of the parties involved in our setting, it is necessary to make a tradeoff between the interests of the individual being photographed and the photographer. As the balance between the right to privacy and the right to photograph, we will now state the minimum rights of each party that should be preserved.

Ideally, the individual should have the right to give consent to every picture she plays a major role in; this is the actual right granted by law in the European Union [140]. This right is hard to enforce technologically, however, as it includes judgment on when a picture is a picture of a person, or just a picture of a marketplace that happens to have people on it. As a minimum, the individual has the right to know she has been photographed, and to have a chance to get an early warning if the picture is being published, which allows her to take appropriate steps in needed.

As long as the photographer does not infringe any personal rights, he should have the right to take pictures without any major obstacles. In this, the protocol should preferably be passive, and not prevent him from taking pictures unless under well defined and measurable circumstances. Furthermore, the photographer has the right to stay anonymous, as long as he does not infringe anybody else's rights. Finally, the photographer has the right to modify his device; for example, the camera in a PDA should not stop working if the operating system is modified or replaced.

5.1.4 Summary of Contributions

With ubiquitous use of digital cameras, e.g. in mobile phones, privacy is no longer threatened by governments and companies only. A new threat exists by people, who take photos of unaware people with no risk and little cost anywhere in public and private spaces. Fast distribution via online communities and web pages expose an individual's private life to the public. Social and legal measures are taken to deal with this, but they are hardly enforceable. In this chapter, we introduce the concept of Personal Rights Management, to protect one's privacy interests in data that is related to that person but held (or owned) by other people. As it is infeasible to enforce a broad ban on camera-phones or artificially inhibit their usage by technical measures, such as with simulated shutter noise, we propose a novel way to complement such measures – attacking the distribution channel. Without eroding the privacy of photographers and putting strong restrictions on cameras, we provide a model for a privacy infrastructure, in which if the picture gets publicly available, the exposed individual has a chance to detect it and take proper actions in the first place.

We exploit several content protection techniques to support the infrastructure. Digital rights management techniques are applied in our proposed infrastructure, and data identification techniques such as digital watermarking and robust perceptual hashing are proposed to enhance the distributed content identification. The implementation with hard- and software solutions of the proposed system are discussed.

5.1.5 Publication Details

The work described in this Chapter has been published in [121, 122].

5.1.6 Chapter Outline

This chapter is organized as follows. The privacy threat is discussed and the attack model is defined, where the attacker and attack scenarios are discussed in Section 5.2.1. The basic protocol on an abstract level is introduced in Section 5.2.2. At the architecture level, Section 5.2.3 proposes a generic evolutionary approach from Digital Rights Management to Personal Rights Management. We propose the protocol based on content identification techniques such as digital watermarking or perceptual image hashing and broadcast channels to enable individuals to take notice when being photographed. After that, Section 5.3 analyzes the hardware infrastructure to implement our protocol, and investigate possible attacks on the hardware. Following this, Section 5.4.1 describes the software implementation of the protocol, both on the side of the camera device and on Internet search engines.

Finally, we discuss the various modifications of the basic scheme in Section 5.5, and draw conclusions towards the feasibility of the technology on mobile phones with particular respect to already existing digital rights management technologies in Section 5.6.

5.2 An Infrastructure for Personal Rights Management

5.2.1 Attack Model

Possible attacks from both the technical and legal aspects will be discussed. From a technical point of view, even with a technically perfect scheme, an attacker could easily circumvent the entire system by using a traditional camera with strong zoom optics or a traditional mini-camera. The problem is not only in the professional voyeurs, but also in the wide deployment of photographic devices and the ease of secret photographing. We assume the attacker can do simple modifications to the device and the picture, and that the corresponding instructions will eventually be published on the Internet. For instance, there are Internet sources to offer modified operating systems for mobile phones to turn off the noise generated while taking a picture. On the other hand, there are many possible attacks for content identification techniques proposed in the literature. However, there is always a balance between the risks for the service provider if the watermarking or hashing scheme is circumvent, and the benefit for the attacker to attempt to break the scheme compare to the amount of effort spent.

From the privacy and legal point of view, it is an unavoidable issue that we want to protect the rights of the harmless photographers. Unless we treat every owner of a mobile phone as a criminal, it would be possible for a sufficiently motivated attacker to escape from the scheme. Apart from making the technology stronger and therefore less attractive to the attackers, our protocol also has its merit if combined with legal measures. By attacking the scheme, it demonstrates a photographer has a “criminal intent”. Therefore, it is easier to distinguish a normally harmless person that just couldn’t resist taking a picture in a particular situation from a semiprofessional voyeur with manipulated equipment.

5.2.2 Basic Protocol

Players

There are three major players in our setting: the photographer, the individual, and the search engine. The *photographer (Bob)* is the person who takes the pictures. Bob uses a camera-phone, which is a mobile phone with a build in camera. From a privacy point of view, Bob has the right not to be inhibited while taking the pictures and has his identity preserved as long as he does not infringe the rights of anybody. Bob also has the right to perform some "standard" changes to his camera-phone, such as updating the operating system.

The *individual (Alice)* is the person that is photographed by the photographer. The interest of Alice is to have control over the pictures taken of her. It means that in the case she is the focal point of the picture, this picture should (ideally) not been taking without her consent. In our protocol, we grant her a lesser right: if a picture taken from her is published, she gets a fair chance to find out early. Alice uses a receiver, which registers the identities of pictures taken in her vicinity. The receiver could be her own mobile phone or a specialized piece of hardware. It can also be integrated in the infrastructure provided by external parties, for instance, the owner of a discotheque or even the GSM operators.

Finally, the *search engine* searches the Internet for picture identities and makes them publicly available. They are similar as any Internet search engines, with slightly modified rules.

Proposed Protocol

A possible scenario of our scheme will be discussed in this section. The goal is to let an individual "Alice" detect unauthorized publication of personal images taken by others "Bob". We name the complete setting a Personal Rights Management (PRM) system.

In the first step, Bob secretly takes private photos of unaware Alice with malicious intent, as shown in Figure 5.1. Luckily, the camera on Bob's mobile phone applies PRM to the photo when it is taken. The photo can be identified and marked by using several data protection techniques, such as digital watermarking, robust perceptual hashing, or Digital Rights Management technology.

In case digital watermarking techniques are used, Bob's camera embeds the image content identification in the picture. In case robust perceptual hashing techniques are used, Bob's camera sends the image hash values optionally together with a thumbnail of the picture itself. All the possible techniques used for content identification will be discussed in the software implementation section.

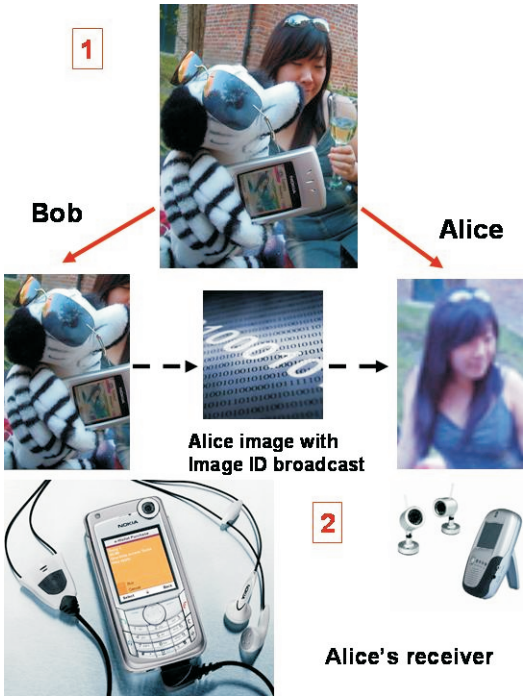


Figure 5.1. The first two steps of the protocol, communication between Alice and Bob. Bob secretly takes private photos of unaware Alice with malicious intent. Alice's image together with identification information are sent to the receiver of Alice

On the other hand, the identity of the photo is broadcast with a short-range radio. Alice's receiver picks up the picture identification information and stores it for later use, as shown in Figure 5.1.

In the next step, Bob publishes the unauthorized photo from Alice to an online community which is very unfavorable to Alice. Alice would take action on this if she knew the photo was published. Luckily, Alice can detect the unauthorized publishing of the photo using the PRM search engine (see Figure 5.2). When Bob puts the picture from Alice on the Internet, the specialized search engines find it and index it by the extracted watermark or the perceptual image hash values. Alice uploads the collected photo marks or identification numbers to a specialized search engine. Then the search engine checks photos published on the web by photo identifications. Upon notification from the search engine, Alice checks whether the photos found have her image on them and takes appropriate actions to protect her privacy. Note that while it is hardly possible to remove data from the Internet if

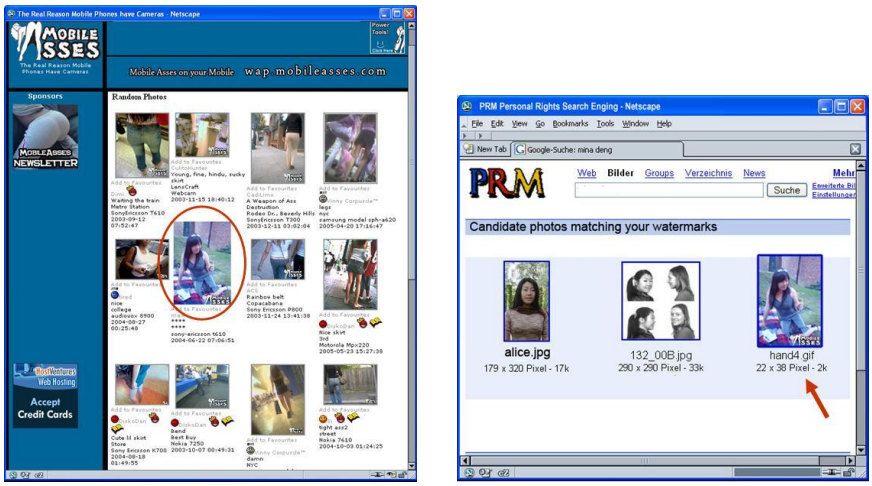


Figure 5.2. The last two steps of the protocol, Bob publishes the unauthorized photo from Alice to an online community which is very unfavorable to Alice. Alice can detect the unauthorized publishing of the photo using the PRM search engine

the publisher is determined to keep it public, we assume that in most real cases the publisher can be identified due to the publishing medium, and convinced to remove the offending material either of the publisher's own accord or by legal means; the possibility for malicious publishers to get away with publishing pictures seems unavoidable unless we want to restrict the privacy of photographers.

In summary, the photo taking is not prevented. From the beginning to the end, Alice and Bob both remain anonymous. Only upon publishing of an image, the image will be detected and reported to Alice.

5.2.3 Architecture Evolution from DRM to PRM

As explained in Section 1.3, the DRM concept can not only be used to protect commercial content, but also be exploited to design systems to protect personal content. At the architecture level, there is a potential of adapting Digital Rights Management (DRM) systems for the purpose of Personal Rights Management (PRM) according to the legal requirements. DRM technology, developed for protecting intellectual property rights, appears to have features that would allow the development of a system-based approach to data protection compliance, i.e. Personal Rights Management.

DRM architectures support description, trading, protection, monitoring and tracking of the use of digital content. These technologies may be contained within

operating systems, program software, or in the hardware of a device.

PRM manages personal data from the data subject, the originator and the owner of the personal data. The EU Data Protection Directive [140] defines the authorities and boundaries of the relationships between each of the participants. The driving purpose behind DRM, thus the content distribution management, relates easily to data protection constructs constraining the exchange of personal data [180].

For the purpose of expressing privacy in a PRM system, the Open Digital Rights Language (ODRL) [168] and extensible rights mark-up language (XrML) [36] can be applied, which are similar as the rights expression languages used in a DRM system.

Korba et al. [180] propose an adaptation of DRM functionality to provide PRM for individuals by assigning names to the functional parts in the DRM setting from the privacy enhancing techniques vocabulary. Thus it brings some form of taxonomy or meta-design for PRM.

5.3 Hardware Implementation

5.3.1 Basic Proposal

The hardware and software implementations of the proposed protocol will be discussed in the following sections. We assume that no mobile phone manufacturer will be willing to add a completely new communication technology into the devices to enable a protocol such as the one presented above. Therefore, we restrict ourselves to the current hardware available in the market. Three possible communication standards, Infrared, Bluetooth, and GSM network, can be used to establish the link between camera-phone from Bob and the receiver from Alice.

Infrared

One feature of infrared communication is that it is directed, i.e., the signal can be sent in a way that only the devices in the view of the camera can receive it. The penalty paid is that the bandwidth of infrared communication is fairly low, and the transmitting distance might be too small. It can cause a problem on the receiving side: if the receiver is not directed to the camera, it may not get any signal at all. It is fairly easy to block the communication by simply gluing an object onto the infrared port. This problem can be solved by building the receiver into the enabling function of camera lens. This way, blocking the communication would disable the ability to take pictures. The second problem could be to block the communication by jamming the signal with a strong infrared light. Though

the problem is harder to deal with, it is possible to design a camera that cannot take pictures if exposed to a strong infrared signal. However, another problem arises when the jamming signal may be directed and allow for a denial of service attack, i.e., preventing all camera phones to take pictures at all.

Bluetooth

Bluetooth communication is the complement of Infrared. The communication is very difficult to jam, and the bandwidth is sufficient even for interactive protocols. The disadvantage is that a Bluetooth signal is undirected and all devices that are not in the visual scope of the camera get the signal as well. Another disadvantage is that currently enabling Bluetooth on a phone may pose a security risk. Recent studies [70] show that many Bluetooth phones are open to attacks that may reveal the entire phone memory, including the address book, the calendar etc. Thus, unless the security of this technique can be improved, to protect the privacy of Alice's pictures she may have to risk a privacy-invasion on her phone book.

GSM-Network

It is by the nature of mobile phones to communicate on the GSM network. However, the GSM protocol is ill-suited for device-to-device communication. Adding this capacity would require major changes in the GSM standard, which is unlikely to happen for the purpose of protecting people from illegal pictures. It would be possible to use the base-station as an intermediate in a way that the photographer's device sends a signal to the base-station, which in turn sends a cell broadcast to all devices in the area. This creates new problems. One on hand, many devices that don't have anything to do with the picture will be noticed altogether. On the other hand, phones at the same location may be locked into another cell or use a different provider.

All of the above

A combination of those techniques can be proposed, for instance, an infrared flash could be used to command the device to listen to a Bluetooth signal or a GSM cellular broadcast. If implemented properly, this could combine the advantages of all technologies. As the infrared signal only has to carry a binary signal, the low bandwidth and limited range are not problematic anymore. As receivers neither see the flash nor listen to the radio signals, they can be configured not to pick up the pictures out of their interests, though it cannot be excluded that they do.

5.3.2 Attacks on the Hardware

A few examples are given here on how an attacker can disable the proposed functionality by manipulating their devices. For some mobile phones, the shutter noise can be manipulated to be turned off when the entire phone is put in silent mode. For our protocol, it is possible to block the transmission by deactivating Bluetooth or by using it to communicate with another device while the picture is taken. Some users directly modify their mobile phone hardware to detach the infrared light or the Bluetooth antenna. For some mobile phones, there is some firmware to manipulate the corresponding functionality available on the Internet. However, mobile phone manufacturers have recently started to think about other functionalities that a user may not manipulate, e.g., Superdistribution and Micropayment from Nokia. It is foreseeable that this problem will be solved in the near future, e.g. by using a core-operating system which cannot be changed by the owner and building the real operating system on top of this core, or by TCPA (Trusted Computing Platform Alliance)/TCG (Trusted Computing Group)-like technologies [280].

5.4 Software Implementation

5.4.1 Digital Image Watermarking

Digital watermarking is a technique for embedding information in digital content without perceptually altering its appearance [102, 60]. In our system, one intuitive way could be to append a visible watermark on the host image. The visible watermark can be any information that identifies the photographer and/or the time stamping analogous to analog cameras. However, the obvious drawback is that an attacker can easily remove the watermark by an image editing software despite of destroying the watermarked region of the image.

Various imperceptible robust image watermarking applications are studied [174, 55, 60]. In the system we proposed, the key point is to identify the secretly photographed image rather than to authenticate the image integrity. This is because Alice is more interested to identify whether the image is from her or not. The owners' and/or user's information can be embedded directly into the images to protect copyright. And a rather high level of robustness against malicious attacks is required.

For watermarking system, it should be computationally infeasible to extract the watermark information even if the algorithm of the watermarking principle is known. Therefore, secret or public keys should be used to provide the security of watermarking.

The design of a watermarking algorithm always involved a tradeoff between robustness, imperceptibility and capacity [197]. In our proposed scheme, the optimal balance among these three attributes should be found if properly designed. The capacity of the watermark does not have to be large, thus extra robustness could be gained. In order to get optimal robustness, watermark should be embedded just below the perceptual level, and the knowledge of human vision systems (HVS) are applied to the imperceptible watermarking schemes [291]. A few benchmarking of watermarking to provide a fair evaluation of watermarking parameters are introduced, such as Stirmark [234], Optimark [220], and Checkmark [232] and so on.

From a practical point of view, with an expected 70 million camera-phones sold by 2006, a 40-bit image identifier should be sufficient even for high usage of the cameras. Although there are no firm numbers, to embed a 40-bit watermark into a picture with 640×480 pixels is quite realistic. For example, the Stirmark [234] can perform the test with 100 bit watermarks on 512×512, 24-bit colored pictures.

When facing a general audience, in order to prevent that everyone has the ability to extract the watermark information from the picture, public key watermarking scheme is used. The photographer embeds the watermark by the public key of the search engine, and the search engine can extract the watermark by using its private key.

5.4.2 Search Engines

The final player of our protocol is a search engine that allows the individual to locate the pictures on the Internet. The search engines could work just like any ordinary one, except for the ability to extract the identification information from the pictures and use it as an index. It requires that the watermark detection or other algorithms to be computationally feasible. Commercial web spiders are already available for copyright protection. As reported in [262], Digimarc, a company which holds most of the core patents on digital watermarking, introduced a tool called MarcSpider [167], purported to crawl the web to search images, test them for watermarks and report on infringers. Due to the fact that crawling the web quickly became an intractable task, as well as that only a small number of copyrighted images installed on the web, MarcSpider didn't work out as a huge success.

Some counter technologies have been developed to hide the pictures from the spider, for example by splitting it into many small pictures or by embedding it using JavaScript. This is another point where a sufficiently motivated attacker can circumvent the scheme, which is hard to deal with unless the privacy of the photographer is inhibited.

There could be a privacy problem introduced by the search engine, such as profiling of all watermarks Alice submits in order to create an album of Alice's life. To avoid linking of ID and image requests, we assume that the search engine is to be used with some anonymous connections.

5.5 Modifications

5.5.1 Perceptual Robust Image Hashing

The watermark-based approach is expected to be sensitive to malicious modifications of the media, thus brings the robustness issue dependent on applications. When the watermark is embedded into the host data, the data content is altered and image manipulations may be localized in most schemes [270].

Robust perceptual hashing, which can be used in multimedia applications both for data identification and robust data authentication, is meant to complement digital watermarking. The main advantage for perceptual hashing schemes is that the data is neither altered nor degraded. If a malicious attack on a watermarking scheme succeeded, the watermark would be destroyed. However, the perceptual hash value will remain the same as long as the perceptual features of the data are unchanged. This is also the reason why perceptual hashing is used instead of cryptographic hashing, which is very sensitive to a single input bit. Perceptual hash functions can be particularly useful to identify illegal copies, since the illegal copies are usually lossy copies of the original.

The main requirement of our scheme is the image identification. An occasional collision between two picture-identities does not cause a significant trouble, although it merely poses a minor annoyance to a user. Therefore, the picture identity does not need to be excessively long. With a k-bit identifier, we need 1.2×2^k pictures to get 0.5 probability of a collision. Therefore, it is proper to apply perceptual hashing schemes to our application.

Four requirements for image hash functions are defined in [205]. A generic image hashing can be achieved in two steps: feature extraction and secure compression of the feature vector. It is shown that the robust feature vector detection is the key point for robust image hashing. Various feature extraction methods are developed based on different concepts, such as by using wavelet [205, 209, 208], DCT [146], matrix invariance [183], different descriptors [206]. The second step that secure compress the feature vector can be based on cryptographic hash functions procedure [270], error correcting codes [205, 209], and secure compression for authentication applications [171].

Having generality and robustness as the two attributes, a feature detection algorithm can be considered robust if it identifies the same feature locations independent of different attacks, such as Stirmark attacks, compression, image processing or geometric distortions. Hamming distances between the hash values of perceptually similar images and between different images can be examined to evaluate the algorithm.

5.5.2 Broadcasting a Sample Picture

In addition to the image identifier, a strongly compressed sample version of the picture could be broadcasted as well. This would inform the individual whether there is a need to take immediate action or not, such as when a specially compromising picture has been taken or a credit card has been photographed. However, this costs a significant bandwidth, and significantly infringes the privacy of the photographer. Due to the fact that the image content taken by the photographer is broadcasted, the photographer could be identified, and therefore, the privacy of the photographer could be violated. Besides, the intellectual property of the photographer, i.e. his work of art in arranging and taking the photo, could be infringed by broadcasting it to the world.

5.5.3 Hybrid DRM Solutions

Several DRM techniques can be integrated into our scheme. In a generic DRM mechanism, digital watermarking and perceptual hashing are used for content protection and/or identification, while encryption and digital signature are used for content confidentiality and integrity [185]. New watermarking based techniques can be used to identify, trace and control the use of digital copy and enhance the content protection, and thus strongly improve DRM [197, 262]. In the application of mobile DRM, watermarking has been suggested as a key technology for *media identification* [282, 159], especially since user's identity is known in mobile networks. Since the market thrives by delivering multimedia content through Multimedia Messaging Service (MMS), the content should be wrapped in DRM packages prior to distribution. The proposed DRM technology for the Open Mobile Alliance (OMA) specifies three different methods that vary in complexity requirements, and that offer different levels of security for the distributed content [48]. The *piracy tracing* with the defense of intellectual property rights and the *copy protection* where a copy-bit is unremovable from the host content [197] require different levels for watermarking robustness.

Encryption and watermarking are to be combined as two defensive lines to enhance DRM. For image content, selective encryption [44] is introduced to encrypt a portion of the compressed data. In our proposed scheme, to protect the

photographer's privacy, the watermarking embedded information can be further encrypted by the user's ID as a secret key, so that only the authenticated party can extract the information [43]. A watermark can be used to serve as a proof of ownership but is vulnerable to attacks such as average and collusion attacks [292]. In addition to ensuring that a watermark cannot be removed, the DRM system has to ensure that a fake watermark cannot be inserted. Several DRM scenarios related to image distributions were analyzed in [197], and a fair and efficient benchmarking of open-source web based evaluation system was proposed. Benchmarking parameters and requirements are scenario dependent.

While discussing the image content protection or identification from a technical perspective, it is important to note that any technique that allows a user to assert their ownership of any digital object must also be placed in the context of intellectual property right law [291].

5.6 Conclusions

Camera-phones have been used in much more malicious ways than just to invade privacy, and control over one's image is hard to enforce today. Several reports have been published of cases where credit card information has been obtained by secret photographing of the card. The problem is analyzed from both the privacy and technical aspects in this chapter, and possible solutions are proposed. There is a tradeoff between the privacy rights of the individual to have control over images and the privacy rights of the photographer. It is of limited effort for initiatives to enact laws to ban the unauthorized photos when lacking of a technological support for the enforcement and prosecution. On the other hand, users and consumers reject technology that presses restrictions on them. While we are aware that our solution – due to the conflicting interests we need to satisfy – leaves a number of issues unresolved, we believe that a great advantage for individual privacy can be achieved by the proposed personal rights management.

This chapter proposes a detection system that combines cryptographic and content protection technologies together with legal enforcements in order to control the distribution of private photos online. The scheme can empower individuals to detect and act upon violations without putting strong restrictions on cameras and photographers. Content identification mechanisms such as digital watermarking and robust perceptual hashing are integrated to enhance a PRM system. Techniques to apply in our scheme are discussed and possible attacks together with hardware and software solutions are analyzed.

To evaluate the usability of our proposed scheme, it is not difficult for one to imagine that it will require a significant amount of time and energy if Alice has to check hundreds of pictures per day from search engines. However, as a normal

individual the chance that Alice gets a high amount of images taken is fairly low. This scheme can be interesting for celebrities though, who are able to afford hiring people to do the checking work in order to make sure that their personal rights are not violated.

Given the potential commercial value of the privacy market, an investment in Personal Rights Management appears to be worthwhile both in terms of what has to be done to achieve compliance with current legislative requirements and to meet privacy policies towards building a stronger trust relationship with customers.

Chapter 6

Conclusions and Future Research

6.1 Conclusions

The goal of this thesis is to investigate privacy issues in content protection systems and to study techniques for privacy preserving content protection. The capabilities of content protection technologies have been criticized for implicating the privacy of content users, by creating the potential for vastly increased collection of information about an individual's intellectual habits and preferences. Therefore, the basic attributes of content protection technologies determine the contradiction between preserving the interests of the content provider or copyright owner for content protection and protecting the privacy of individuals.

The research questions addressed in this thesis span two dimensions. Primarily, the research aims to lay out groundwork for a privacy threat analysis and requirement engineering methodology. Although digital privacy is one of the identified priorities in our society, few systematic, effective methodologies exist to deal with privacy threats thoroughly. We present a comprehensive framework to model privacy threats for supporting the elicitation and fulfilment of privacy requirements in application systems. The second part of the research aims to tackle the privacy protection issues in a number of designated content protection systems. New privacy threats emerge when limited trust is put on the content or the service provider. Therefore, the proper balancing between content protection, for the content provider or service provider, and privacy protection, for the user, remains a research challenge. The privacy preserving content protection systems proposed in this thesis can be categorized into two types. The first is privacy preserving DRM

techniques to protect commercial content, and the proposed solution is anonymous buyer-seller watermarking protocol. The other is to design systems that protect personal content using content protection techniques. Such a system targets the preservation of individuals' privacy interests when personal content is held by other parties. The proposed systems include a privacy friendly architecture to manage distributed e-Health content, and a personal rights management system to enforce individual privacy rights. The conclusions of this thesis are outlined in the rest of the section.

Several conclusions can be drawn from the research. First, the research shows that privacy, just as user behavior regulation, has become one of the values embodied in content protection system design. In addition, the development of content protection technologies can respond to privacy protection requirements in a goal-oriented approach, such as following the proposed framework for privacy threat and requirement analysis, while complying with relevant legislation. Moreover, instead of impeding privacy, content protection technologies can be utilized to preserve and protect it.

6.1.1 Privacy Threats and Requirements Framework

In the first part of the research, we laid out a comprehensive and generic framework to model privacy threats to elicit privacy requirements and instantiate privacy enhancing countermeasures. The purpose of this taxonomic framework is to aid the development of a privacy enhancing system. As presented in Chapter 2, a number of fundamental questions have been investigated, including conceptualizing privacy properties and threats, evaluating the relation between privacy and security properties, and designing a generic methodology to identify privacy threats and elicit privacy requirements in application-dependent systems.

Conceptualizing Hard Privacy & Soft Privacy Properties. The presented work further deepens the understanding of privacy as *hard privacy* and *soft privacy*, first introduced by George Danezis [109, 111], by concretely defining privacy properties (and threats) according to these two concepts. First, the taxonomy of privacy properties in a broader context is proposed, taking the aforementioned dualism into consideration; it extends the privacy terminology proposed by Pfitzmann and Hansen [237]. Hard privacy properties include unlinkability, anonymity and pseudonymity, plausible deniability, undetectability and unobservability, and confidentiality. Soft privacy properties are proposed as content unawareness and policy and consent compliance.

Modeling Privacy Threats in a System. One of the primary contributions of the work is a methodology to model privacy specific threats in an application

system. This is achieved by defining a list of privacy threat types and providing the necessary mappings to the elements in the system model, presented as data flow diagram for instance. The taxonomy of privacy threats is derived from the privacy properties, including linkability, identifiability, non-repudiation, detectability and observability, information disclosure, content unawareness, and policy and consent noncompliance. The proposed systemic approach to privacy threat modeling is named *LINDDUN*, each letter of which stands for a privacy threat obtained by negating a privacy property.

Instantiating Privacy Threats via Threat Tree Patterns. The second contribution is represented by the supporting body of knowledge – an extensive catalogue of privacy specific threat tree patterns – as a guideline to detail the generic LINDDUN threat categories into specific threat instances that can occur in a system. We presented a significant number of threat tree patterns to detail the privacy threats in a system, as an illustrative indication, allowing system analysts to consider the most common privacy conditions. These privacy threat trees, inspired by security threat tree patterns in Secure Development Lifecycle (SDL), are based on the state-of-art privacy developments and are susceptible to a continuous improvement process based on newly discovered threats. During the risk-analysis phase, some threat instances might be discarded. Thereafter, misuse cases are instantiated regarding the result of the above process as a collection of threat scenarios that need to be documented. A misuse case, in particular, can be viewed as a use case from an attacker's point of view.

Selecting Privacy Enhancing Countermeasures. Another contribution is the provision of the means to map the most commonly known privacy enhancing technologies (PETs) to the identified privacy threats and elicited privacy requirements of a system. Given that both threats and privacy technologies addressing them are in constant evolution, the privacy threat tree patterns and categorization of suggested PETs are expected to be continuously updated and improved upon. Leveraging the link between privacy enhancing technologies and privacy properties, it is possible to make a distinction between hard and soft privacy enhancing solutions. Hard privacy technologies are actively researched but inadequate in deployment, due to cost and technical restrictions. Soft privacy technologies are the-state-of-art and have fewer research activity. With legal compliance as a strong driver, soft privacy solutions rely on the stakeholder's liability and the tradeoffs between the cost of deploying privacy solutions and the potential costs in case of massive data breaches.

6.1.2 Privacy Preserving Content Protection Systems

The second part of our research focuses on exploring and designing individual privacy friendly content protection systems. We follow the fundamental assumption that the content provider or the service provider is not trustworthy and hence new privacy threats emerge. This research question is addressed from two viewpoints: (1) to design content protection systems with privacy preserving properties for protecting commercial content, and (2) to design privacy preserving systems to manage and protect personal data using content protection systems or techniques.

Anonymous Watermarking Protocols. To address the privacy issue in DRM-supported systems for online distribution of commercial content, anonymous buyer-seller watermarking protocols have been introduced. Their goal is to resolve the conflicting interests between copyright protection of the content, such as the content providers' need for copyright protections and traceability for the copyright violator, and privacy protection, such as anonymity (or pseudonymity) and the unlinkability of online transactions of customers. We defined the fundamental requirements of such watermarking protocols, proposed and constructed three types of anonymous buyer-seller watermarking (BSW) protocols based on homomorphic encryption and group signatures. In contrast to earlier work, the proposed anonymous BSW protocols fulfill the desired security properties simultaneously. In addition, we provided a formal security definition for generic copyright protection protocols in the ideal-world/real-world paradigm. Furthermore, we have analyzed the security of an anonymous buyer-seller watermarking protocol and proven that it fulfills our definition. In particular, we have shown that the protocol is secure against any p.p.t. (probabilistic polynomial-time) adversary when instantiated with a watermarking scheme, an encryption scheme, a group signature scheme and zero-knowledge proofs of knowledge that provide security against any p.p.t. adversary. Unlike the other building blocks, however, no watermarking scheme has been proven to offer this security level, and thus the actual security of the protocol against malicious buyers is lowered to the security offered by the watermarking scheme.

An efficient implementation of the aforementioned buyer-seller watermarking protocol has been evaluated. The implementation combines existing cryptographic tools with a composite signal representation in the encrypted domain, allowing the reduction of both the computational overhead and the large communication bandwidth introduced by the use of homomorphic public-key cryptosystems. Considering the computational and network capacity of modern systems, the results suggest that the proposed technique can be successfully used in practical applications in the near future. More detailed implementation results are given in [115, 116, 117].

Privacy Enhancing Sharing of Distributed E-Health Information. A privacy friendly architecture to manage distributed personal e-health information has been presented in this thesis, in order to address the contradiction between privacy protection for the patient and the sharing of medical data. On the one hand, the primary goal of an e-Health system is to provide a patient-centric lifelong view on the medical data under conditional access. On the other hand, e-Health data have a number of distinguishing features. For example, they are regarded as sensitive data and, according to European data protection legislation, cannot be processed or distributed without the patient's consent. This raises two specific privacy threats in a distributed e-Health system. Firstly, in order to retrieve all the necessary medical data of a patient, there must be a mechanism to cross-reference medical documents across several healthcare providers. This enables, for instance, patient profiling by data aggregation. Another privacy threat occurs, for instance, when a patient's medical data is transferred from one healthcare provider (such as a psychiatric hospital) to another healthcare provider (such as a generic hospital), where not all the information should be disclosed to the latter provider due to the different privacy sensitivity levels of medical data.

To mitigate these two privacy threats, a privacy protection mechanism is proposed, with a limited trust on the healthcare and service providers, in order to facilitate a privacy enhancing sharing of distributed e-Health information. In the proposed setting, each patient is issued with a unique context-specific identifier by each healthcare provider. In communications across different healthcare providers (or contexts), the same information can be expressed by means of different types or values. To accommodate the conflicting forces between protecting a patient's ID and personal e-Health information against unauthorized parties (even those within healthcare providers), and facilitating the possibility to follow-up a patient's medical treatment history by authorized parties, we introduce the concept of the mediating service to map and convert context-specific identifiers or information, when data is exchanged among different authorized healthcare providers. Further, we ensure that only the minimum necessary patient's personal information is provided to the authorized parties, such that the protocol is compliant with the data minimization principle. This is accomplished by introducing a data anonymization mechanism to obfuscate sensitive parts according to the patient's privacy preference. This user-centric approach satisfies both interoperability and privacy protection.

Personal Rights Management. The broad usage of hand-held devices such as camera-phones has resulted in far more threats than just the demise of privacy, and control over personal content such as one's image is hard to enforce today. For instance, several reports have been published of cases where credit card information has been obtained by secretly photographing the card. To address this issue, we introduced the concept of Personal Rights Management to protect

one's privacy interests in data that is related to that person but held by other people. We sketched out a detection mechanism that combines cryptographic and content protection technologies, in compliance with legal regulations, in order to control the distribution of private photos online. The proposed infrastructure targets the distribution channel. As soon as personal content such as a picture is publicly available, the exposed individual has a chance to find it and take proper action in the first place, without putting strong restrictions on cameras and photographers. The implementation issues with hardware and software solutions of the proposed system are discussed. Digital rights management techniques are applied in our proposed infrastructure, and data identification techniques such as digital watermarking and robust perceptual hashing are proposed to enhance the distributed content identification.

6.1.3 Insights into the Design of Privacy Preserving Systems

During the course of the research, we have gained insights into the design of privacy enhancing systems. The observations are derived from the various research challenges investigated in each chapter, and are refined and generalized in the rest of this section. The range of topics this section touches is broad, and we can only scratch the surface here. Much further work is needed to develop these concepts as understanding of privacy evolves. We still hope that our observations can inspire system designers and privacy engineers.

The Mutually Inclusive Relationship Between Privacy and Security. Our first observation is the mutually inclusive relationship between privacy and security. There is an intersection between privacy properties and security properties. This leads to two observations. On the one hand, privacy can be partially viewed as a top tier security property, rather than a security tradeoff. The means and ends of information privacy and information security are essentially similar. At first, the concept of self-determination for privacy is considered as the most valued security property. In addition, privacy satisfies some fundamental security needs of individuals, such as freedom from surveillance and profiling, freedom from compulsion, and accessibility to content and services. Excessive violation of privacy undermines security. Furthermore, both depend critically on technologies, while legal enforcement is necessary but insufficient.

On the other hand, the pluralistic meaning of privacy (explained in Section 1.1.3) determines that some privacy properties are not included in the category of security properties. The goals of privacy and security can be in conflict. One example is, as explained in Section 2.4.5, depending on the application, plausible deniability (as a privacy property) can at times be desirable over non-repudiation (as a security property). This contradictory phenomenon can be understood as follows. When referring to convincing a particular party (within a

system) about the existence of a particular event, non-repudiation and plausible deniability are mutually exclusive. However, the two properties can coexist within the whole system, in which some parties retain evidence of an event while others don't, or the evidence is sufficient to convince some parties but not others.

Privacy in View of Relationships. In the process of conceptualizing privacy properties, another interesting observation leads us to draw the conclusion that privacy emphasizes relationships between entities and data. In other words, privacy cares more about the relationship between instances of the system DFD (data flow diagram) elements (including entities, data flows, data stores, and processes), for example, the relationship of two data flows (i.e., two communication instances cannot be linked), and the relationship between a DFD element with an entity (i.e., a person sends a message anonymously). On the contrary, security focuses more on the functionality of each individual DFD component (i.e., an entity, a data flow, a data store, and a process) from a more isolated view.

Privacy and Security Analysis in Synergy. We pointed out the synergy between the privacy analysis and security threat analysis, and we recommend analyzing both privacy and security threats and requirements together as one complete process. An example is, when taking privacy threat trees, one can see many privacy threats leading to security threats (e.g. information disclosure and tampering). This implies that, privacy objectives often depend on security objectives (e.g. confidentiality and integrity of data flow, data store or process). Three motivations support this statement. First, the aforementioned fact confirms that security is a necessary means to achieve privacy. Second, both the proposed LINDDUN framework (in Chapter 2) for privacy and the STRIDE framework for security can be well integrated into the Security Development Lifecycle. This synergy brings advantages in terms of time and cost for system designers to work with. Third, in spite of the coexistence of security and privacy in one system, security objectives might conflict with privacy objectives (e.g. non-repudiation and plausible deniability as explained in the previous paragraphs). It is thus useful to do the threat analysis and consider requirements both for security and privacy together.

Integrated Approach to Build in Privacy. A high-performance architectural design process to build in privacy cannot be achieved unless an integrated design approach is deployed. Privacy, just like security, is part of the entire design-and-build process. Privacy cannot be achieved solely through individual components without considering the system as a whole. In the integrated design approach for building in privacy, the privacy objective needs to be identified right at the start of the design process, and held in proper balance among the stakeholders during the design process. Later in the planning and implementation phase, the relationships and interdependencies of the privacy components with all the

other components in the system should be concurrently evaluated, appropriately applied and coordinated. In the meanwhile, the effectiveness and impact of privacy measures must be constantly reassessed to ensure the coordination of the privacy objective with all the components in the system. This assessment does not only address the privacy outcomes when the system is working correctly, but also considers the harm to the system caused by potential failures.

Balancing Privacy with Cost and Efficiency. Good privacy or security engineering does not always mean to provide perfect or nearly perfect solutions. Instead, it needs to find a proper balance among several tradeoffs, while taking the system's practical constraints into consideration. One tradeoff is between privacy and efficiency; the other is between privacy and cost. These two tradeoffs are interactive. Generally speaking, the more privacy enhancing a system is, the less efficiently it works, and the more costly it is. The deployment of privacy enhancing solutions is usually at the price of increasing the implementation budget or lowering the performance efficiency of the system. One observation is that mathematically proven secure and private systems tend to be more complex and costly than those with a lesser degree of security and privacy. Take the example of the anonymous buyer-seller watermarking protocols we developed (in Chapter 3). Adding zero-knowledge proofs to such a cryptographic protocol generally intends to offer designated privacy or security properties, at the expense however, of degraded implementation efficiency.

One pragmatic approach to address this issue is through cost-benefit analysis. The cost of building in privacy can be evaluated both in terms of the research and development cost (e.g. investment in R&D projects) and in terms of the potential degradation of system performance (e.g. efficiency reduction). The cost of building in no privacy can be evaluated, for example, in terms of direct and indirect financial loss and reputation damage caused by potential privacy breaches in the system. Overall, building in privacy or security could be an option as long as it costs less than building in no privacy or security.

With balancing between cost and efficiency being one challenge for privacy research, designing more efficient systems at a lower cost, while still preserving designated privacy and security properties, remains another research challenge.

Both Technical and Legal Enforcement. Our last part of conclusion is that one needs both technological and legal mechanisms to ensure privacy protection. This means that, on the one hand, legal enforcement plays an important role. Assuming secret information was disclosed, one needs to rely on law or policy that shapes the potential future behavior around the exposed information. On the other hand, relying only on legal enforcement does not suffice; technology is an indispensable means for privacy assurance. There are already a large number of privacy enhancing technologies that effectively ensure direct privacy preservation. However, despite technology insuring privacy with high confidence, legal enforcement is still necessary to ensure the technological enforcement is in compliance with legislation.

As a summary, we provide a few lines of insights into privacy engineering. (1) Privacy threats can elicit requirements, and requirements must specify privacy objectives. (2) After requirements are determined, they should be implemented. (3) Comparable with security, privacy needs to be technologically supported, whereas privacy legislation is necessary but insufficient to protect privacy.

6.2 Future Work

Building privacy in content protection systems is an emerging and challenging research topic that has attracted an increasing amount of research activities. This thesis has laid the general groundwork for instantiating privacy threats to support the elicitation and fulfilment of privacy requirements in an application-dependent systems. Moreover, three kinds of privacy preserving content protection systems for both commercial content and personal data have been explored. The following section will bring forward an assessment of the most prevalent and promising directions for future research.

Privacy Framework.

Practical Validation. As future work, the proposed framework can be applied to larger scale case studies. Although not discussed in this thesis, a validation in the context of a national e-health system has been performed [127]. Future work includes the privacy threat analysis of the proposed buyer-seller watermarking protocols and the personal rights management system using the privacy framework.

Risk Assessment. Risk assessment to privacy threats is another challenge, not only because of its academic value but also the industrial relevance. Intuitively speaking, it could be infeasible to assign a specific risk level to a certain privacy threat when generalizing all types of systems. However, we believe it is a valid approach to provide risk priority classifications for a particular type of system (such as social networks, e-health systems, anonymous communication systems, etc.) on a case-by-case basis. One of the methods to achieve risk assessment is, for instance, by exploiting privacy metrics. This risk assessment will merely be useful as a guideline because the actual risk levels will still depend on the designated system's requirements.

Automated Design. Another direction to complete the privacy framework is to develop software design programs to enable an automated process for instantiating privacy threats and requirements. Privacy requirements can be elicited in a goal-oriented approach, e.g. using a requirements modeling language proposed by Yu et al. [298].

Watermarking Protocols.

New Properties. Further research can be conducted to adapt or extend our definition of anonymous buyer-seller watermarking protocols that offer additional properties. For example, one desirable property for e-commerce protocols is transaction fairness [184], and thus defining and designing privacy-preserving fair buyer-seller watermarking protocols is an interesting goal.

Hardware Implementation. Another research challenging is to implement the protocol in hardware, such as a portable device, bringing the potential to be released on markets as a commercial product.

Distributed E-Health Systems.

User-Centricity – Patient Specification. E-health systems are evolving towards user-centric systems, in which the patients are able to control the granularity of healthcare information disclosed to third parties, and specify the content of the health information and to which healthcare provider it can be disclosed, the purpose of processing the information, etc.

Transparency – Patient Verification. Transparency needs to be emphasized more in e-Health systems such that the patients should be able to access and query the logs of their health records, in order to verify if their records were accessed according to the rules they have defined.

API Interoperability. Application Programming Interface (API) is used to specify how an individual e-Health system works within each healthcare provider. So far there is no interaction between different APIs. The next generation e-Health systems should be able to offer collaboration between healthcare providers by using a data bus interconnecting different service providers, so that each API is not only available but also used for the exchange of health information. The healthcare information stored in the local database is transferred and translated by the data bus and can be shared between two or more healthcare providers. This interoperability can be realized by implementing the interoperable APIs across different healthcare providers.

Personal Rights Management.

Software and Hardware Implementation. The general concept of Personal Rights Management is designed to keep protection as well as to track the sharing process of personal data. Based on the PRM concept to control personal images as we proposed in this thesis, further research can be focused on working out the protocol prototype implementations and security. For example, TCPA/TCG-

like [280] trusted computing platforms and DRM systems could be integrated into the prototype to achieve a generic PRM architecture.

Efficiency Improvements. We discussed the time problem if Alice has to check hundreds of pictures per day. We propose to ease the problem by adding location and biometric (e.g. facial) data recognition algorithms to search engines to reduce the complexity for Alice. Future research could address this feature's implementation.

New Applications. New applications of PRM could be expanded into other aspects of peer-to-peer privacy violations. While private image taking is the most eminent area of privacy issues caused by peers, other threats are emerging. There is a vast increase of video camera-phones on the market, which brings a similar privacy threat as the privacy image scenario. There are many mp3 players equipped with recording functions. There is a tendency to upload electronically recorded conversations or videos online, and the presence of a high number of uncontrolled recording devices may pose a significant problem in the future. A recent story of a high school teacher Jay Bennish in the US shows an example of a problem caused by privately owned recording equipment. The teacher's speech was investigated, because of a student's recording in class, and complained to the principal [218]. Another emerging problem is the ever-increasing number of blogs and web sites similar as YouTube.com, combined with search engines to efficiently find personal information therein. Furthermore, PRM scenarios could be applied to protect personal geographical location data as well.

Commercial Value. Given the potential commercial value of the privacy market, an investment in Personal Rights Management appears to be worthwhile both in terms of what has to be done to achieve compliance with current legislative requirements and to meet privacy policies towards building a stronger trust relationship with clients.

Appendix A

Privacy Misuse Case Examples

A.1 MUC 2: Linkability of the User-Portal Data Stream (Data Flow)

Summary: Data flows can be linked to the same person (without necessarily revealing the person's identity)
Asset: PII of the user

- The user:
    1. data flow can be linked to each other which might reveal the person's identity
    2. the attacker can build a profile of a user's online activities (interests, active time, comments, updates, etc.)

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor intercepts / eavesdrops two or more data flows
2. The misactor can link the data flows to each other and possibly link them (by combining this information) to the user / data subject

Trigger: by misactor, can happen whenever data is communicated
Preconditions:

1. No anonymous communication system used
2. Information disclosure of data flow possible

Prevention capture points:

1. Use strong anonymous communication techniques
2. Provide confidential channel

Prevention guarantee: Impossible to link data to each other

A.2 MUC 3: Linkability of the Social Network Users (Entity)

Summary: Entities (with different pseudonyms) can be linked to the same person (without necessarily revealing the person's identity)
Asset: PII of the user

- The user:
    1. data can be linked to each other which might reveal the person's identity
    2. attacker can build a profile of a user's online activities (interests, actives time, comments, updates, etc.)

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor intercepts or eavesdrops two or more pseudonyms
2. The misactor can link the pseudonyms to each other and possibly link (by combining this information) to the user / data subject

Trigger: by misactor, can happen whenever data is communicated
Preconditions:

1. Information Disclosure of the data flow possible

2. Different "pseudonyms" are linked to each other based on content of the data flow

Prevention capture points:

1. protection of information such as user temporary ID, IP address, time and location, session ID, identifier and biometrics, computer ID, communication content, e.g. apply data obfuscation to protection this information (security)
2. message and channel confidentiality provided

Prevention guarantee: Impossible to link data to each other

A.3 MUC 4: Identifiability at the Social Network Database (Data Store)

Summary: The user's identity is revealed
Asset: PII of the user

- The user: revealed identity

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor gains access to the database
2. The data is linked to a pseudonym
3. The misactor can link the pseudonym to the actual identity (identifiability of entity)
4. The misactor can link the data to the actual user's identity

Alternative Flow:

1. The misactor gains access to the database
2. The can link information from the database to other information (from another database or information which might be publicly accessible)

3. The misactor can re-identify the user based on the combined information

Trigger: by misactor, can always happen
Preconditions:

1. no or insufficient protection of the data store
2. no data anonymization techniques used

Prevention capture points:

1. protection of the data store (security)
2. apply data anonymization techniques

Prevention guarantee: hard-impossible to link data to identity (depending on applied technique)

A.4 MUC 5: Identifiability of the User-Portal Data Stream (Data Flow)

Summary: The user's identity is revealed
Asset: PII of the user

- The user: revealed identity

Primary misactor: insider / outsider
Basic Flow:

1. The misactor gains access to the data flow
2. The data contains personal identifiable information about the user (user relationships, address, etc.)
3. The misactor is able to extract personal identifiable information from the user / data subject

Trigger: by misactor, can happen whenever data is communicated
Preconditions:

1. no or weak anonymous communication system used
2. Information disclosure of data flow possible

Prevention capture points:

1. apply anonymous communication techniques
2. Use confidential channel

Prevention guarantee: hard-impossible to link data to identity (depending on applied technique)

A.5 MUC 6: Identifiability of Social Network System Users (Entity)

Summary: The user's identity is revealed
Asset: PII of the user

- The user: revealed identity

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor gains access to the data flow
2. The data contains the user's password
3. The misactor has access to the identity management database
4. The misactor can link the password to the user

Alternative Flow:

1. The misactor gains access to the data flow
2. The data contains the user's password
3. The misactor can link the user's password to the user's identity (e.g. passwords as initials followed by birthdates)

Trigger: by misactor, can happen whenever data is communicated and the user logs in using his "secret"
Preconditions:

1. Insecure IDM system OR
2. weak passwords used and information disclosure of data flow possible

Prevention capture points:

1. Strong pseudonymity technique used (e.g. strong passwords)
2. privacy-enhancing IDM system
3. Data flow confidentiality

Prevention guarantee: hard(er) to link log-in to identity.

A.6 MUC 7: Information Disclosure at the Social Network Database (Data Store)

Summary: Data is exposed to unauthorized users
Asset: PII of the user

- The user: revealed sensitive data

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor gains access to the database
2. The misactor retrieves data to which he should not have access

Trigger: by misactor, can always happen
Preconditions:

1. no or insufficient internal access policies

Prevention capture points:

1. strong access control policies (security). For example, rule-based access control based on friendships in the social network

Prevention guarantee: hard-impossible to obtain data without having the necessary permissions

A.7 MUC 8: Information Disclosure of the User Data Stream (Data Flow)

Summary: The communication is exposed to unauthorized users
Asset: PII of the user

- The user: revealed sensitive data

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor gains access to the data flow
2. The misactor retrieves data to which he should not have access

Trigger: by misactor, can happen whenever messages are being sent
Preconditions:

1. communication goes through insecure public network

Prevention capture points:

1. messages sent between user and social network web client is encrypted and secure communication channel is ensured

Prevention guarantee: hard-impossible to gain access to the data flow without having the right permissions

A.8 MUC 9: Content Unawareness

Summary: User is unaware that his or her anonymity is at risk due to the fact that too much personal identifiable information is released
Asset: PII of the user

- The user: revealed identity

Primary misactor: skilled insider / skilled outsider
Basic Flow:

1. The misactor gain access to user's online comments
2. The misactor profiles the user's data and can identify the user

Trigger: by misactor, can always happen
Preconditions:

1. User provides too much personal data

Prevention capture points:

1. User provides only minimal set of required information

Prevention guarantee: user will be informed about potential privacy risks

A.9 MUC 10: Policy and Consent Noncompliance

As explained in Section 2.5.2, the policy and consent noncompliance threat affects the system as a whole (including data flow, data store and process). We only illustrate one of the misuse cases for the policy and consent noncompliance threat in the rest of the section.

Summary: The social network provider doesn't process user's personal data in compliance with user consent, e.g., disclose the database to third parties for secondary use
Asset: PII of the user

- The user: revealed identity and personal information
- The system / company: negative impact on reputation

Primary misactor: Insider
Basic Flow:

1. The misactor gains access to social network database
2. The misactor discloses the data to a third party

Trigger: by misactor, can always happen
Preconditions:

1. misactor can tamper with privacy policies and makes consents inconsistent OR
2. policies not managed correctly (not updated according to user's requests)

Prevention capture points:

1. Design system in compliance with legal guidelines for privacy and data protection and keep internal policies consistent with policies communicated to user
2. Legal enforcement: user can sue the social network provider whenever his or her personal data is processed without consents
3. Employee contracts: employees who share information with 3th parties will be penalized (fired, pay fine, etc.)

Prevention guarantee: Legal enforcement will lower the threat of an insider leaking information but it will still be possible to breach user's privacy

Appendix B

Security analysis of Type III BSW protocol

Theorem 1. *This BSW scheme securely realizes $\mathcal{F}_{DRM}$.*

In order to prove this theorem, we need to build a simulator $\mathcal{E}$ that invokes a copy of adversary $\mathcal{A}$ and interacts with $\mathcal{F}_{\text{DRM}}$ and environment $\mathcal{Z}$ in such a way that ensembles $\text{IDEAL}_{\mathcal{F}_{\text{DRM}},\mathcal{E},\mathcal{Z}}$ and $\text{REAL}_{\text{DRM},\mathcal{A},\mathcal{Z}}$ are computationally indistinguishable.

We analyze formally the security of our scheme when the seller and a subset of buyers are corrupted, and when (a subset of) buyers are corrupted. We also describe briefly the security guarantees that our scheme provides when the registration authority and the deanonymization authority are corrupted.

B.1 Security Analysis When Seller Is Corrupted

Claim 1. *When the seller and a subset of the buyers are corrupted, the distribution ensembles* $\text{IDEAL}_{\mathcal{F}_{DRM},\mathcal{E},\mathcal{Z}}$ *and* $\text{REAL}_{DRM,\mathcal{A},\mathcal{Z}}$ *are computationally indistinguishable under the zero-knowledge property of the proofs of knowledge, the IND-CPA security of encryption schemes* $(\mathsf{BKeygen}, \mathsf{BEnc}, \mathsf{BDec})$ *and* $(\mathsf{JKeygen}, \mathsf{JEnc}, \mathsf{JDec})$*, and the traceability, non-frameability and anonymity properties of the group signature scheme.*

Proof. We show by means of a series of hybrid games that the environment $\mathcal{Z}$ cannot distinguish between the real execution ensemble $\text{REAL}_{\text{DRM},\mathcal{A},\mathcal{Z}}$ and the

simulated ensemble $\text{IDEAL}_{\mathcal{F}_{\text{DRM}},\mathcal{E},\mathcal{Z}}$ with non-negligible probability. We denote by Pr [**Game** i] the probability that $\mathcal{Z}$ distinguishes between the ensemble of **Game** i and that of the real execution.

- **Game** 0: This game corresponds to the execution of the real-world protocol with a subset of honest buyers and honest $\mathcal{J}$, $\mathcal{R}$ and $\mathcal{D}$. Thus Pr [**Game** 0] = 0.

- **Game** 1: This game proceeds as **Game** 0, except that **Game** 1 aborts if the received message-signature pair (m, s_m) is correct according to algorithm GSverify but cannot be successfully opened through algorithm GSopen. The probability that **Game** 1 aborts is bounded by the following lemma:

  Lemma 1. *Under the traceability property of the group signature scheme,* $|Pr\,[\textbf{\textit{Game}}\ 1] - Pr\,[\textbf{\textit{Game}}\ 0]| = \nu_1$.

  Proof. We construct an algorithm $\mathcal{T}$ that, if there exists an adversary $\mathcal{A}$ that makes **Game** 1 abort with non-negligible probability ϵ, breaks the traceability property of the group signature scheme with non-negligible probability ϵ. The traceability property is formally defined in [66] as a game between a challenger $\mathcal{C}$ and an adversary. First, $\mathcal{C}$ gives to the adversary (gpk, osk) and access to several oracles (we refer to [66] for the description of the oracles). Eventually, adversary submits a message-signature pair (m, s_m), and wins the game if $\mathsf{GSverify}(gpk, m, s_m)$ outputs 1 and if $\mathsf{GSopen}(gpk, osk, reg, m, s_m)$ outputs a pair $(i, proof)$ such that either $i = 0$ or $\mathsf{GSjudge}(gpk, i, upk_i, m, s_m, proof)$ outputs 0.

  Algorithm $\mathcal{T}$ operates as follows. First, $\mathcal{T}$ receives (gpk, osk) from $\mathcal{C}$ and sends gpk to $\mathcal{A}$ when queried with (crs). For each honest buyer $\mathcal{B}_i$, $\mathcal{T}$ invokes oracle $\mathsf{AddU}(i)$ and later on oracle USK to obtain the secret key usk_i and the private signing key gpk_i. Each time $\mathcal{A}$ wants to register a public key upk_i of a corrupted buyer $\mathcal{B}_i$, $\mathcal{T}$ invokes the corruption oracle $\mathsf{CrptU}(i, upk_i)$. When $\mathcal{A}$ sends a request m to register a corrupted buyer $\mathcal{B}_i$, $\mathcal{T}$ invokes oracle $\mathsf{SndTol}(i, m)$. $\mathcal{T}$ simulates purchase requests by honest buyers following algorithm Request. Each time $\mathcal{A}$ sends an arbitration message $(\phi, W_{\mathcal{S}}, m, s_m)$, $\mathcal{T}$ runs $\mathsf{GSopen}(gpk, osk, reg, m, s_m)$ to obtain i and $proof$. If either $i = 0$ or $\mathsf{GSjudge}(gpk, i, upk_i, m, s_m, proof)$ outputs 0, $\mathcal{T}$ sends (m, s_m) to break the traceability property. □

- **Game** 2: This game proceeds as **Game** 1, except that **Game** 2 aborts if, in the arbitration phase, $\mathcal{A}$ sends a message-signature pair (m, s_m) that algorithm GSopen opens successfully to an uncorrupted buyer's identity i and buyer $\mathcal{B}_i$ did not send a signature on m to $\mathcal{A}$. The probability that $\mathcal{Z}$ distinguishes between **Game** 2 and **Game** 1 is bounded by the following lemma:

Lemma 2. *Under the non-frameability of the group signature scheme,* $|Pr\,[\boldsymbol{Game}\ 2] - Pr\,[\boldsymbol{Game}\ 1]| = \nu_2$.

Proof. We construct an algorithm $\mathcal{T}$ that, if there exists an adversary $\mathcal{A}$ that makes **Game** 2 abort with non-negligible probability ϵ, breaks the non-frameability property of the group signature scheme with non-negligible probability ϵ. The non-frameability property is formally defined in [66] as a game between a challenger $\mathcal{C}$ and an adversary. First, $\mathcal{C}$ gives to the adversary (gpk, isk, osk) and access to several oracles (we refer to [66] for the description of the oracles). Eventually, adversary submits a message-signature pair (m, s_m) and a proof $(i, proof)$, and wins the game if $\mathsf{GSverify}(gpk, m, s_m)$ outputs 1, if i belongs to an honest user and if $\mathsf{GSjudge}(gpk, i, upk_i, m, s_m, proof)$ outputs 1.

Algorithm $\mathcal{T}$ operates as follows. First, $\mathcal{T}$ receives (gpk, isk, osk) from $\mathcal{C}$ and sends gpk to $\mathcal{A}$ when queried with (crs). Each time $\mathcal{A}$ wishes to register a public key upk_i of a corrupted buyer $\mathcal{B}_i$, $\mathcal{T}$ invokes oracle $\mathsf{CrptU}(i, upk_i)$. Each time $\mathcal{A}$ wishes to register a corrupted buyer, $\mathcal{T}$ runs $\mathsf{GSiss}(gpk, isk, upk_i)$ with $\mathcal{A}$. For every honest buyer $\mathcal{B}_i$, $\mathcal{T}$ invokes oracle $\mathsf{SndToU}(i, \cdot)$ and stores the output. For each purchase request made by an honest buyer $\mathcal{B}_i$ for item j, $\mathcal{T}$ computes a request message m following algorithm $\mathsf{Request}$, obtains a signature s_m by invoking oracle $\mathsf{GSig}(i, m)$ and sends (m, s_m) to $\mathcal{A}$. Each time $\mathcal{A}$ sends an arbitration message $(\phi, W_{\mathcal{S}}, m, s_m)$, $\mathcal{T}$ runs $\mathsf{GSopen}(gpk, osk, reg, m, s_m)$ to get i and $proof$. If i belongs to an honest buyer, $\mathcal{A}$ did not receive before a signature by i on m, and $\mathsf{GSjudge}(gpk, i, upk_i, m, s_m, proof)$ outputs 1, then $\mathcal{T}$ sends $(m, s_m, i, proof)$ to $\mathcal{C}$ to break the non-frameability property. □

- **Game** 3: This game proceeds as **Game** 2, except that the proofs $\pi_1 = \mathsf{PK}\{(sk_{\mathcal{B}}{}') : (pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}') \leftarrow \mathsf{BKeygen}(1^k) \wedge C \leftarrow \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}{}')\}$ and $\pi_{2i} = \mathsf{PK}\{(W_{\mathcal{B}i}) : c_i \leftarrow \mathsf{BEnc}(pk_{\mathcal{B}}{}', W_{\mathcal{B}i}) \wedge W_{\mathcal{B}i} \in \{0,1\}\}$ are replaced by simulated proofs. Under the assumption that the proof system is zero-knowledge, $|\Pr\,[\textbf{Game}\ 3] - \Pr\,[\textbf{Game}\ 2]| = \nu_3$.

- **Game** 4: This game proceeds as **Game** 3, except that the ciphertext $C = \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}{}')$ is replaced by a ciphertext that encrypts a random message. At this point, the proof of knowledge $\pi_1 = \mathsf{PK}\{(sk_{\mathcal{B}}{}') : (pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}') \leftarrow \mathsf{BKeygen}(1^k) \wedge C \leftarrow \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}}{}')\}$ is a simulated proof of a false statement. The probability that $\mathcal{Z}$ distinguishes between **Game** 4 and **Game** 3 is bounded by the following lemma:

Lemma 3. *Under the IND-CPA security of the encryption scheme that consists of algorithms* $(\mathsf{JKeygen}, \mathsf{JEnc}, \mathsf{JDec})$, $|Pr\,[\boldsymbol{Game}\ 4] - Pr\,[\boldsymbol{Game}\ 3]| = \nu_4$.

Proof. We construct an algorithm $\mathcal{T}$ that, given an environment $\mathcal{Z}$ that distinguishes **Game** 4 and **Game** 3 with non-negligible probability, breaks the IND-CPA security of the encryption scheme with non-negligible probability. Chosen plaintext security is formally defined through a game between a challenger $\mathcal{C}$ and an adversary [155]. First, $\mathcal{C}$ provides the adversary with a public key pk. The adversary sends two messages m_0 and m_1. $\mathcal{C}$ flips a coin b and sends $C = \mathsf{Enc}(pk, m_b)$ to the adversary. Finally, the adversary sends his guess b' and wins if $|\Pr[b = b'] - \frac{1}{2}|$ is non-negligible.

Let N be the number of purchase requests. We consider a sequence of hybrid games, where, in game-j, ciphertext C is replaced by the encryption of a random message in the first j purchase requests, while the remaining requests remain unchanged. Clearly, game-0 corresponds to **Game** 3 and game-N corresponds to **Game** 4. If $\mathcal{Z}$ distinguishes **Game** 4 and **Game** 3 with non-negligible probability ϵ, there must be an index j such that $\mathcal{Z}$ distinguishes game-j from game-$(j+1)$ with non-negligible probability ϵ/N.

Our algorithm $\mathcal{T}$ operates as follows. First, $\mathcal{T}$ receives the public key pk from $\mathcal{C}$. $\mathcal{T}$ computes (gpk, isk, osk) by running GSgkg and sends gpk to $\mathcal{A}$ when queried with (crs). $\mathcal{T}$ registers adversarial buyers as usual. $\mathcal{T}$ computes $(pk_{\mathcal{B}}{}', sk_{\mathcal{B}}{}')$. For $i = 1$ to j, it computes purchase requests following algorithm $\mathsf{Request}$, except that C is replaced by the encryption of a random value and π_1 by a simulated proof. For $i = j + 2$ to N, purchase requests are computed following algorithm $\mathsf{Request}$. For $i = j + 1$, $\mathcal{T}$ picks random m and submits $(sk_{\mathcal{B}}{}', m)$ to $\mathcal{C}$. $\mathcal{C}$ flips a coin b and returns $C = \mathsf{JEnc}(pk, m_b)$, and uses C to compute the request. $\mathcal{Z}$ outputs a bit b', which is forwarded by $\mathcal{T}$ to $\mathcal{C}$. □

- **Game** 5: This game proceeds as **Game** 4, except that **Game** 5 aborts upon receiving an arbitration request $(W_{\mathcal{B}}, m, s_m)$ where (m, s_m) was previously sent to $\mathcal{A}$ and $W_{\mathcal{B}}$ was the buyer's watermark associated with the request (m, s_m). The probability that $\mathcal{Z}$ distinguishes between **Game** 5 and **Game** 4 is bounded by the following lemma:

  Lemma 4. *Under the IND-CPA security of the encryption scheme that consists of algorithms* $(\mathsf{BKeygen}, \mathsf{BEnc}, \mathsf{BDec})$, $|Pr\,[\textbf{\textit{Game}}\ 5] - Pr\,[\textbf{\textit{Game}}\ 4]| = \nu_5$.

  Proof. Let N be the number of purchase requests. We construct an algorithm $\mathcal{T}$ that, given an adversary $\mathcal{A}$ that makes **Game** 5 abort with non-negligible probability, breaks the chosen plaintext security of the encryption scheme with non-negligible probability ϵ/N.

  Algorithm $\mathcal{T}$ operates as follows. First, $\mathcal{T}$ receives the public key $pk_{\mathcal{B}}$ from $\mathcal{C}$. $\mathcal{T}$ computes (gpk, isk, osk) by running GSgkg and sends gpk to $\mathcal{A}$ when queried with (crs). $\mathcal{T}$ registers adversarial buyers as usual. For the first

purchase request made by an honest buyer $\mathcal{B}_i$ for item j, $\mathcal{T}$ picks random $m \leftarrow \{0,1\}^{l_2-1}$ and, for $i = 1$ to $l_2 - 1$, encrypts bitwise m using $c_i = \mathsf{BEnc}(pk_{\mathcal{B}}{}', m_i)$. To encrypt the last bit, $\mathcal{T}$ sends $(0,1)$ to $\mathcal{C}$ and receives back a ciphertext c, which is used to complete the bitwise encryption of the buyer's watermark $W_{\mathcal{B}}$. The rest of the request message is computed following algorithm Request, except that the encryption $C = \mathsf{JEnc}(pk_{\mathcal{J}}, sk_{\mathcal{B}})$ is replaced by the encryption of a random value and the proofs π_1 and π_{2i} are replaced by simulated proofs. (Note that $\mathcal{T}$ knows neither $sk_{\mathcal{B}}$ nor the bit encrypted in c.) The remaining $N-1$ requests are computed following algorithm Request. $\mathcal{A}$ sends an arbitration message $(W_{\mathcal{B}}, m, s_m)$ that makes **Game** 5 abort. If this arbitration message does not correspond to the first request, $\mathcal{T}$ fails. Otherwise, if the last bit of $W_{\mathcal{B}}$ is 0, $\mathcal{T}$ sends $b' = 0$ to $\mathcal{C}$, and otherwise $b' = 1$ to $\mathcal{C}$. □

- **Game** 6: This game proceeds as **Game** 5, except that all the group signatures of purchase requests are replaced by group signatures computed by using the same private signing key of a unique buyer. The probability that $\mathcal{Z}$ distinguishes between **Game** 6 and **Game** 5 is bounded by the following lemma:

  Lemma 5. *Under the anonymity property of the group signature scheme,* $|Pr\,[\textbf{\textit{Game}}\ 6] - Pr\,[\textbf{\textit{Game}}\ 5]| = \nu_6$.

  Proof. We note that, at this point, we have already proven that $\mathcal{A}$ is not able to frame honest buyers, who by assumption do not release pirated copies. Therefore, the identity of an honest buyer will never be revealed at the arbitration protocol, and so the change we make on the identity of the buyer that computes purchase requests cannot be detected there. We only have to prove that this change is indistinguishable at the purchase phase.

  The anonymity property of dynamic group signatures is formally defined in [66] and it consists of a game between a challenger $\mathcal{C}$ and an adversary. First, the challenger gives the adversary (gpk, isk) and access to several oracles. Then adversary gives the challenger a message m and two identities i_0 and i_1. $\mathcal{C}$ flips a coin b and sends to adversary a group signature $s = \mathsf{GSsig}(gsk_{i_b}, m)$. $\mathcal{A}$ wins if he guesses b with non-negligible probability.

  We employ a sequence of hybrid games. Let game-0 denote the game in which all the group signatures remain unmodified, and game-N denote the game in which all of them have been replaced. Clearly, game-0 corresponds to **Game** 5 and game-N corresponds to **Game** 6. If there is an environment $\mathcal{Z}$ that distinguishes **Game** 6 and **Game** 5 with non-negligible probability ϵ, then there exists an index j such that $\mathcal{Z}$ distinguishes game-j and game-$(j+1)$ with non-negligible probability ϵ/N. Given such $\mathcal{Z}$, we construct an algorithm $\mathcal{T}$ that breaks the anonymity property of the group signature

scheme with non-negligible probability ϵ/N. Our algorithm $\mathcal{T}$ receives (gpk, isk) from $\mathcal{C}$. $\mathcal{T}$ invokes oracle $\mathsf{SndToU}(i', \cdot)$ to register the new honest user i' employed to simulate purchase requests. $\mathcal{T}$ follows algorithm $\mathsf{Request}$ to compute the request message m. Then $\mathcal{T}$ sends (m, i, i'), where i is the identity of the original buyer $\mathcal{B}_i$ that sends the request, as its challenge. $\mathcal{C}$ flips a coin b returns a signature $s_m = \mathsf{GSsig}(gsk_{i_b}, m)$ of m, and $\mathcal{T}$ sends (m, s_m) to $\mathcal{A}$. If $b = 0$, the distribution corresponds to game-j, and, if $b = 1$, to game-$(j + 1)$. $\mathcal{Z}$ outputs a bit b', which is forwarded by $\mathcal{T}$ to challenger as its guess. □

$\mathcal{E}$ performs all the changes described in **Game** 6, and forwards and receives messages from $\mathcal{F}_{\text{DRM}}$ as described in our simulation below:

- **Setup.** When $\mathcal{A}$ sends a request (crs) to obtain gpk, $\mathcal{E}$ runs GSgkg to obtain the group public key gpk, the issuer's secret key isk and the opening secret key osk. $\mathcal{E}$ sends (crs, gpk) to $\mathcal{A}$. When $\mathcal{A}$ sends a request $(\text{retrieve}, \mathcal{J})$, $\mathcal{E}$ runs $\mathsf{JKeygen}$ in order to generate a key pair $(pk_{\mathcal{J}}, sk_{\mathcal{J}})$ and sends $(\text{retrieve}, \mathcal{J}, pk_{\mathcal{J}})$ to $\mathcal{A}$.
- **Registration.** Upon receiving a registration request from $\mathcal{A}$, $\mathcal{E}$ executes the interactive algorithm GSiss on input (gpk, isk, upk_i). If the execution ends successfully, $\mathcal{E}$ stores reg_i in reg and sends (register) to $\mathcal{F}_{\text{DRM}}$ on behalf of $\mathcal{B}_i$. $\mathcal{E}$ knows the identity $\mathcal{B}_i$ of the corrupted buyer because the communication channel is authenticated.
- **Purchase.** Upon receiving $(\text{buyrequest}, j)$ from $\mathcal{F}_{\text{DRM}}$, if this is the first request $\mathcal{E}$ runs GSukg to obtain a user key pair (usk, upk) and algorithms GSiss and GSjoin on input (gpk, isk, upk) and (gpk, usk) respectively to obtain a private signing key gsk. This key is used to simulate all the requests. $\mathcal{E}$ follows the interactive algorithm $\mathsf{Request}(gpk, gsk_i, j, pk_{\mathcal{J}})$ with all the changes described until **Game** 6 to compute a request for item j and receive watermarked content Y. $\mathcal{E}$ stores the request (m, s_m) along with $W_{\mathcal{B}}$ in the request table T_{req} and sends $(\text{reqresp}, Y)$ to $\mathcal{F}_{\text{DRM}}$.
- **Release.** Upon receiving a pirated copy Y' from $\mathcal{A}$, $\mathcal{E}$ sends $(\text{release}, Y')$ to $\mathcal{F}_{\text{DRM}}$ and stores Y' in a table T_{rel} of released copies.
- **Arbitration.** When $\mathcal{A}$ sends $(info)$, $\mathcal{E}$ parses $info$ as $(W_{\mathcal{B}}, m, s_m)$, verifies s_m and checks if m encrypts $W_{\mathcal{B}}$. If it is not the case, $\mathcal{E}$ sends $(\text{detect}, \perp)$ to $\mathcal{F}_{\text{DRM}}$, receives $(\text{detresp}, \textit{not guilty})$ and forwards $(\text{detresp}, \textit{not guilty})$ to $\mathcal{F}_{\text{DRM}}$. Otherwise $\mathcal{E}$ runs $\mathsf{GSopen}(gpk, osk, reg, m, s_m)$ and obtains an identifier i and a proof $proof$. ($\mathcal{E}$ aborts if $(W_{\mathcal{B}}, m, s_m)$ fulfills any of the conditions described in the sequence of games.) Then $\mathcal{E}$ proceeds as follows:

– If i corresponds to an adversarial buyer, $\mathcal{E}$ chooses any of the pirated copies $Y \in T_{rel}$ and sends (detect, Y) to $\mathcal{F}_{\text{DRM}}$. $\mathcal{F}_{\text{DRM}}$ returns (detresp, $\mathcal{A}$, *guilty*), which is forwarded to $\mathcal{A}$.
– If i corresponds to the buyer used by $\mathcal{E}$ to simulate purchases, $\mathcal{E}$ sends (detect, $\perp$) to $\mathcal{F}_{\text{DRM}}$. (Note that we assume that honest buyers never release pirated copies.) $\mathcal{F}_{\text{DRM}}$ returns (detresp, *not guilty*), which is forwarded to $\mathcal{A}$.

The distribution produced in **Game** 6 is identical to that of our simulation. By summation we have that $|\Pr[\textbf{Game } 6] \leq \nu_7$. □

B.2 Security Analysis When Buyers Are Corrupted

Claim 2. *When only (a subset of) the buyers are corrupted, the distribution ensembles* $\text{IDEAL}_{\mathcal{F}_{DRM},\mathcal{E},\mathcal{Z}}$ *and* $\text{REAL}_{DRM,\mathcal{A},\mathcal{Z}}$ *are computationally indistinguishable under the traceability and non-frameability properties of the group signature scheme and the collusion resistance of the watermarking scheme.*

Proof. We show by means of a series of hybrid games that the environment $\mathcal{Z}$ cannot distinguish between the real execution ensemble $\text{REAL}_{\text{DRM},\mathcal{A},\mathcal{Z}}$ and the simulated ensemble $\text{IDEAL}_{\mathcal{F}_{\text{DRM}},\mathcal{E},\mathcal{Z}}$ with non-negligible probability.

- **Game** 0: This game corresponds to the execution of the real-world protocol with honest $\mathcal{S}$, $\mathcal{J}$, $\mathcal{R}$ and $\mathcal{D}$. Therefore, $\Pr[\textbf{Game } 0] = 0$.
- **Game** 1: This game proceeds as **Game** 0, except that **Game** 1 aborts if the received message-signature pair (m, s_m) is correct but cannot be opened through algorithm GSopen. The probability that **Game** 1 aborts is bounded by the following lemma:

  Lemma 6. *Under the traceability property of the group signature scheme,* $|Pr[\textbf{Game } 1] - Pr[\textbf{Game } 0]| = \nu_1$.

  The proof of this lemma follows the proof of traceability given in Appendix B.1.
- **Game** 2: This game proceeds as **Game** 1, except that **Game** 2 aborts if the received message-signature pair (m, s_m) is opened correctly to an uncorrupted buyer's identity i. The probability that **Game** 2 aborts is bounded by the following lemma:

  Lemma 7. *Under the non-frameability property of the group signature scheme,* $|Pr[\textbf{Game } 2] - Pr[\textbf{Game } 1]| = \nu_2$.

The proof of this lemma follows the proof of non-frameability given in Appendix B.1.

- **Game** 3 : This game operates as **Game** 2, except that the string $W = \phi||(W_{\mathcal{S}} \oplus W_{\mathcal{B}})$ that is used to compute the watermark embedding is replaced by a random string. Since the strings ϕ and $W_{\mathcal{S}}$ are picked at random by the honest seller, W is a random string that leaks no information on $W_{\mathcal{B}}$. Therefore, $|\Pr[\textbf{Game } 3] - \Pr[\textbf{Game } 2]| = 0$.

- **Game** 4 : This game operates as **Game** 3, except that **Game** 4 aborts if $\mathcal{A}$ releases a watermarked content Y whose watermark W does not equal that of any of the watermarked content previously received by $\mathcal{A}$. The probability that **Game** 4 aborts is bounded by the following lemma:

  Lemma 8. *Under the assumption that the watermarking scheme is collusion resistant,* $|Pr[\textbf{\textit{Game } 4}] - Pr[\textbf{\textit{Game } 3}]| = \nu_3$.

  We construct an algorithm $\mathcal{T}$ that, given an adversary $\mathcal{A}$ that makes **Game** 4 abort with non-negligible probability, breaks the collusion resistant property of the watermarking scheme with non-negligible probability. $\mathcal{T}$ interacts with the challenger $\mathcal{C}$ of the collusion resistant game described in Definition 1. First, $\mathcal{T}$ receives the challenge $(Y_1, \ldots, Y_l)$ from $\mathcal{C}$. $\mathcal{T}$ computes (gpk, isk, osk) by running GSgkg and sends gpk to $\mathcal{A}$ when queried with (crs). $\mathcal{T}$ registers adversarial buyers as usual. When receiving a purchase request for item 1, $\mathcal{T}$ replies by encrypting a not previously used Y_i with $pk_{\mathcal{B}}{}'$. (We assume that item 1 is requested no more than l times.) For other purchase, $\mathcal{T}$ replies as usual. Eventually, $\mathcal{A}$ releases a pirated copy Y' whose watermark does not equal any of the watermarks embedded in $(Y_1, \ldots, Y_l)$. $\mathcal{T}$ forwards Y' to $\mathcal{C}$.

$\mathcal{E}$ performs all the changes described in **Game** 4, and forwards and receives messages from $\mathcal{F}_{\text{DRM}}$ as described in our simulation below.

- **Setup.** When $\mathcal{A}$ sends a request (crs) to obtain gpk, $\mathcal{E}$ runs GSgkg to obtain the group public key gpk, the issuer's secret key isk and the opening secret key osk. $\mathcal{E}$ sends (crs, gpk) to $\mathcal{A}$. When $\mathcal{A}$ sends a request $(\text{register}, upk_i)$ to register the public key upk_i of buyer $\mathcal{B}_i$, $\mathcal{E}$ stores $(\mathcal{B}_i, upk_i)$. When $\mathcal{A}$ sends a request $(\text{retrieve}, \mathcal{J})$, $\mathcal{E}$ runs $\mathsf{JKeygen}$ in order to generate a key pair $(pk_{\mathcal{J}}, sk_{\mathcal{J}})$ and sends $(\text{retrieve}, \mathcal{J}, pk_{\mathcal{J}})$ to $\mathcal{A}$.

- **Registration.** Upon receiving a registration request from $\mathcal{A}$, $\mathcal{E}$ behaves as in Appendix B.1.

- **Purchase.** Upon receiving (m, s_m) from $\mathcal{A}$, $\mathcal{E}$ checks whether $\mathsf{GSverify}(gpk, m, s_m)$ is correct. As verifier, $\mathcal{E}$ executes the proofs π_1 and, for $i = 1$ to l_2,

Requirement	Opener	Issuer
Anonymity	uncorrupt	fully corrupt
Traceability	partially corrupt	uncorrupt
Non-frameability	fully corrupt	fully corrupt

Table B.1. Levels of trust in authorities for each security property

π_{2i}, and ignores the request if any of them fails. $\mathcal{E}$ runs $(i, \pi) \leftarrow \mathsf{GSopen}$ (gpk, osk, reg, m, s_m), parses m as $(pk_{\mathcal{B}}{}', j, (c_i)_{i=1}^{l_2}, C)$ and sends $(\text{request}, j)$ to $\mathcal{F}_{\text{DRM}}$ on behalf of $\mathcal{B}_i$. $\mathcal{F}_{\text{DRM}}$ returns $(\text{reqresp}, Y)$. $\mathcal{E}$ computes $ct = \mathsf{BEnc}(pk_{\mathcal{B}}{}', Y)$ and sends ct to $\mathcal{A}$.

- **Release.** Upon receiving a pirated copy Y' from $\mathcal{A}$, $\mathcal{E}$ sends $(\text{release}, Y')$ to $\mathcal{F}_{\text{DRM}}$.

The distribution produced in **Game** 4 is identical to that of our simulation. By summation we have that $|\Pr[\textbf{Game } 3] \leq \nu_4$. □

B.3 Security Analysis When Other Parties Are Corrupted

We do not formally analyze the security of our scheme in these cases since in practical application scenarios the registration authority $\mathcal{R}$ and the deanonymization authority $\mathcal{D}$ are trusted. We note that the security of our scheme relies on the security of the group signature scheme. In our scheme, $\mathcal{R}$ acts as the issuer of the group signature scheme, and $\mathcal{D}$ acts as the opener. Bellare et al. [66] analyze the security of the group signature scheme when the adversary corrupts the issuer and the opener. In Table B.1, they describe the maximum level of corruption that the scheme tolerates so that anonymity, traceability and non-frameability still hold. (Partial corruption means that the secret key of a party is revealed to the adversary, but the adversary cannot influence the behavior of that party.) Interestingly, non-frameability holds even when the issuer and the opener are fully corrupted. Therefore, our scheme protects honest buyers from being falsely accused when $\mathcal{S}$, $\mathcal{R}$ and $\mathcal{D}$ are corrupted. We recall that we assume that the judge is always uncorrupted.

Appendix C

Implementation of Type III BSW protocol

This section briefly discusses the implementation of the proposed Type III BSW protocol. The protocol efficiency is first verified by estimating the complexity of the different parts of the protocol, considering well-known practical implementation designs for the cryptographic primitives employed therein. Then, realistic performance measures are derived by running a practical implementation of the buyer-seller watermarking protocol on a network of general purpose personal computers. Results on the efficient implementations of using the composite embedding strategy are presented in [115, 116, 117]. All tested programs have been implemented in C++ using the GNU Multi-Precision (GMP) library [1] and the NTL library [6].

C.1 Efficiency Analysis

We estimate the protocol efficiency in terms of computational and communication complexity for realistic parameters. For the computational complexity, the total number of exponentiations required by the protocols, with the group size on which they are performed, are presented in Table C.1. The communication complexity is evaluated as the sum of the sizes of all messages or rounds, i.e., the number of bits exchanged during the protocols. Based on the same group, we distinguish single exponentiations (denoted as *exp.*) with multi-exponentiations (denoted as *multi.*), taking into consideration that there are algorithms to compute multi-exponentiations that are faster than first computing each exponentiation separately and then multiplying the results.

Table C.1. Computational complexity and communication complexity estimation of the Type III BSW protocol

Protocol	number of exp. or multi-exp. (group size)	size (bit)
Protocol 1	(2 exp.+ 4 multi.) (on 2048 bits)	12,853
(embedding)		
(–pixelwise)	(262,144 multi. (on 2048 bits))	(536,870,912)
(–composite)	(3,760 multi. (on 2048 bits))	(6,318,080)
(extraction)		
(–pixelwise)	(262, 144 multi. (on 2048 bits)	0
(–composite)	(3, 760 multi. (on 2048 bits))	0
Protocol 2		
–pixelwise	18 exp. (on 1024 bits), (1,158 exp.+ 524,427 multi.) (on 2048 bits)	538,028,566
–composite	18 exp. (on 1024 bits), (1,158 exp.+ 7,659 multi.) (on 2048 bits)	7,475,734
Protocol 3	(3 exp.+101 multi.) (on 2048 bits)	429,320
Total		
–pixelwise	18 exp. (on 1024 bits), (1,163 exp. + 524,532 multi.) (on 2048 bits)	538,470,739
–composite	18 exp. (on 1024 bits), (1,163 exp. + 7,764 multi.) (on 2048 bits)	7,917,907

In Table C.1 we consider a 512×512 image with a size of roughly $512 \times 512 \times 8 \approx 2$ Mbits, so that the size of the host signal is 262,144 pixels, with a watermark (or fingerprint) of 128 bits, of which 96 bits of the watermark are generated by the buyer and the seller and 32 bits are used for the index. When considering the step of the watermark embedding in the encrypted domain, with the pixelwise approach, each pixel is encrypted using Paillier's cryptosystem, requiring 262,144 multi-exponentiations on a 2048-bit group. The size of the encrypted image is $262,144 \times 2048 = 536,870,912$ bits. When using the efficient composite signal representation [115, 116], we assume that the quantization scale factor $Q = 2^{11}$, so that we have roughly 3,760 multi-exponentiations on a 2048-bit group and 6,318,080 transmitted bits.

From Table C.1, it is evident that the total number of exponentiations is dominated by the number of multi-exponentiations, and that the most of the computational effort is required to encrypt and decrypt the whole image. A great amount of the computational complexity is located on the seller's side, since the seller has to encrypt the digital content and perform the embedding in the encrypted domain.

However, the composite signal representation can significantly lower this burden. In the pixelwise case, the number of exponentiations required to encrypt and decrypt the image takes 99.95% of the total number of exponentiations on a 2048-bit group, whereas in the efficient composite representation case this ratio is reduced to 96.86%. As to the communication efficiency, the transmission of the encrypted image takes 99.7% of the bandwidth in the pixel wise case and 79.8% of the bandwidth in the composite case. The data also shows that the overhead of the protocol is small compared to image encryption: to protect a $512 \times 512 \times 8 = 2$ Mbits image, the data exchanged in the whole protocol (composite version) is about 7.9 Mbits with the composite embedding. With an expansion rate of 3.95, relatively small compared to most public key cryptosystems, and considering the modern network bandwidth capacity, we can conclude that the communication overhead is within an acceptable range.

C.2 Protocol Implementation

C.2.1 Watermark Embedding

The first step of the protocol implementation is to evaluate the performance gain obtained by the composite embedding strategy versus the pixelwise embedding strategy.

In the following paragraph, we show the result of two implementation strategies of the watermark embedding and extraction. The first version is to encrypt and decrypt each pixel separately. This strategy is referred to as *pixelwise*. The second version employs the composite signal representation [115, 116] and is referred to as *efficient composite*. For instance, with the order of the composite representation $R = 85$, every 85 pixels of the image are encrypted to a ciphertext.

The aforementioned versions have been run on an Intel®Core(TM)2 Quad CPU at 2.40 GHz, used as a single processor. Preliminary results of the comparison can be found in [115, 116], where we measured the execution time of both versions using various image sizes. In each version, a random bit sequence with the same length (i.e. 128 bits) as the watermark has been embedded using the Quantization Index Modulation (QIM) [97] watermarking technique. Both implementation strategies are based on the Paillier's cryptosystem [230] with a modulus N of 1024 bits. The implementation results for two different image sizes – 512×512 and 1024×1024 – are depicted in Table C.2.

It is evident that the composite signal representation permits reducing the computational complexity of secure watermark embedding to a great extent. Namely, when the quantization scale factor is $Q = 2^{11}$, the execution time of the efficient composite embedding is 70 times less than the pixelwise embedding,

Table C.2. Execution times (in seconds) of the two implementation strategies of the watermarking embedding and extraction algorithm: pixelwise and efficient composite

(in seconds)	512×512		1024×1024	
	pixelwise	composite	pixelwise	composite
embedding	2,058	30	7,528	110
extraction	546	7	2,171	28

and the corresponding extraction operation is about 80 times faster with respect to the pixelwise version. A 1024×1024 image can be processed by the seller in less than two minutes, whereas the buyer can extract the plaintext image in less than 30 seconds. For a 512×512 image, the watermark embedding by the seller takes 30 seconds, and the buyer can extract the plaintext image in 7 seconds. Such timing constraints do not seem prohibitive in view of a practical application of the proposed techniques.

C.2.2 Complete Protocol

Based on the aforementioned preliminary results, we further tested a complete implementation of the proposed protocol. The implementation consists of a set of four programs, each implementing a different entity of the protocol. The $\mathcal{S}$, the buyer $\mathcal{B}$, and the judge $\mathcal{J}$ are implemented as separate programs. The functionalities of the registration authority $\mathcal{R}$ and the deanonymization authority $\mathcal{D}$ are implemented in a single program. The programs communicate with each other via TCP (Trusted Computing Platform), using the standard socket library provided by the Linux operating system.

The buyer $\mathcal{B}$ and the judge $\mathcal{J}$ have been tested on an Intel®Core(TM)2 Quad CPU at 2.40 GHz, used as a single processor. The seller $\mathcal{S}$ has been tested on an AMD Athlon 64 at 2.40 GHz. The registration/deanonymization authority has been tested on an Intel®Centrino(TM) at 1.7 GHz. The machines were connected by a high-speed LAN. We tested two different image sizes, 512×512, and 1024×1024, and two different watermark lengths, 64 and 128 bits. In order to investigate the effects of security parameters on the complexity, since the number of group signature operations is negligible with respect to the number of Paillier's encryptions, we only change the encryption security parameters. Four different security levels for Paillier's cryptosystem were considered, using keys with 512, 1024, 2048, and 3072 bits, whereas the group signature scheme used 2048 bit keys. In order to minimize the complexity, we used the efficient composite embedding strategy with a quantization scale factor $Q = 2^{11}$.

Table C.3. Execution times (in seconds) of the seller and the buyer in the watermark generation and embedding phase (WGE), zero knowledge proof for fair encryption of private keys (π_1), zero knowledge proof for bit encryption (π_2), watermark embedding (only computation time) and extraction of watermarked image (only computation time): (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

(in milliseconds)	Seller		Buyer	
	overall	computation	overall	computation
WGE	68,973	4,7911	79,159	14,372
π_1	873	68	962	32
π_2	25,684	5,908	24,853	3,088
Embedding	–	39,474	–	–
Extraction	–	–	–	8,376

For each entity we measured the exchanged bytes, the actual computation time, and the overall execution time that is the sum of the actual computation time and the handshake latency. For the Watermark Generation and Embedding Protocol, the execution times for the seller and the buyer are given in Figure C.1 and Figure C.2. For the Identification and Arbitration Protocol, the execution times for the seller, the judge, and the deanonymization authority are given in Fig. C.3, Figure C.4, and Figure C.5. The latencies in Figure C.3 and Figure C.5 occur because the seller and the deanonymization authority have to wait for the judge's reply in the arbitration phase. For the registration protocol, the overall execution time is always below 500 milliseconds, so its complexity is negligible with respect to the other phases.

When taking the following parameters: for an 512×512 image, the watermark size is 128 bits, and Pailliar's key size is 1024 bits, Table C.3 depicts the execution times (in seconds) of the seller and the buyer in the watermark generation and embedding phase (WGE), zero knowledge proof for fair encryption of private keys π_1, zero knowledge proof for bit encryption π_2, watermark embedding (only computation time) and extraction of watermarked image (only computation time). In the watermarking phase (WGE), the computation time of the buyer is less than the seller (less than one third). The latency is due to the fact that the buyer must wait for the seller to complete the protocol. The computation times for the watermark embedding (only the seller) and the extraction of the watermarked image (only for the buyer) are comparable to the preliminary results shown in Table C.2. From table C.3, we can see that the zero knowledge proof π_1's complexity is negligible with respect to the other protocols, since all the execution times are less than 1 second. The zero knowledge proof π_2 has a lot of handshake latency because it is repeated 128 times.

From the above figures, it is evident that the most computationally demanding part is the Watermark Generation and Embedding Protocol. The overall complexity is dominated by the computation time of the seller, which is about four times higher than the computation time of the buyer. It is worth noting how the different parts of the protocol contribute to the overall complexity. From Figure C.6, Figure C.7, and Figure C.8, we can see that most of the computation time of the seller is devoted to the watermark embedding, while the complexity of the buyer is equally affected by the computation time of the watermarked image extraction and the round complexity of the zero knowledge proof π_2. However, since the complexity of the zero knowledge proof π_2 does not depend on the image size, with larger images the overall complexity appears to be dominated by the watermark embedding and extraction parts. Except for the zero knowledge proof π_2 and the protocol executed by the judge, the overall complexity of the other parties is independent from the length of the watermark. As to the computation time versus the length of Paillier's key, we can notice a quadratic law: this is in agreement with the fact that exponentiations on k bits have an $O(k^2)$ complexity.

Finally, the total amount of data exchanged in the Watermark Generation and Embedding Protocol and in the Identification and Arbitration Protocol are shown in Figure C.9 and Figure C.10. The communication complexity of the verification phase is considerably low. Furthermore, it is interesting to note that, because of the composite representation, the communication complexity of the embedding protocol does not depend on the length of Paillier's key.

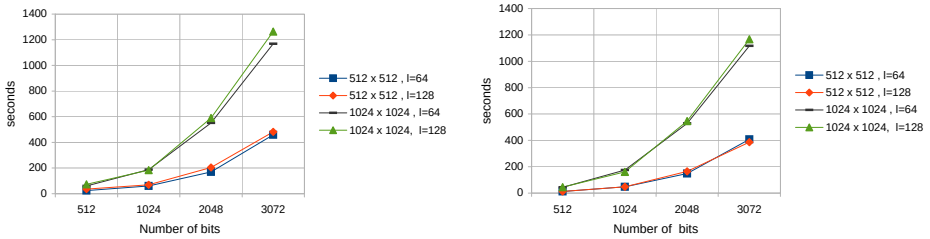


Figure C.1. Execution times (in seconds) of the seller in the Watermark Generation and Embedding Protocol versus the number of bits of Paillier's key: (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

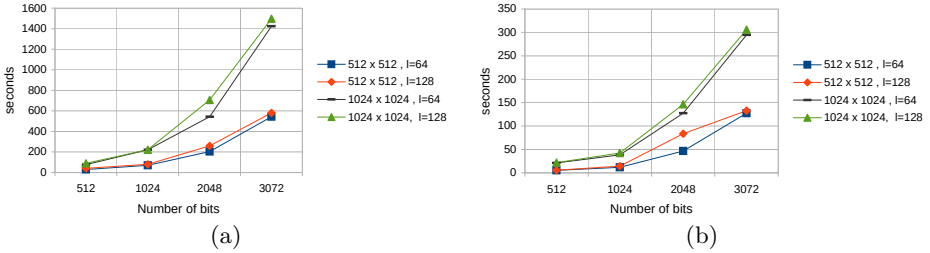


Figure C.2. Execution times (in seconds) of the buyer in the Watermark Generation and Embedding Protocol versus the number of bits of Paillier's key: (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

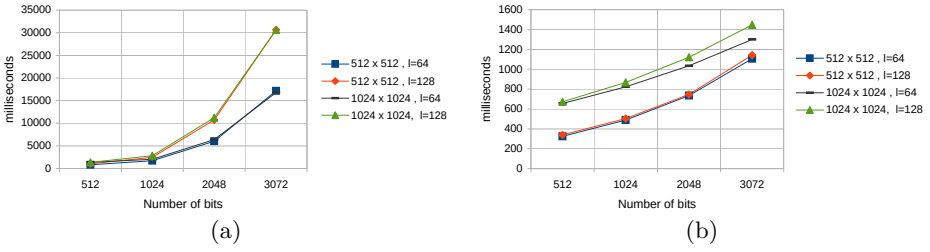


Figure C.3. Execution times (in milliseconds) of the seller in the Identification and Arbitration Protocol versus the number of bits of Paillier's key: (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

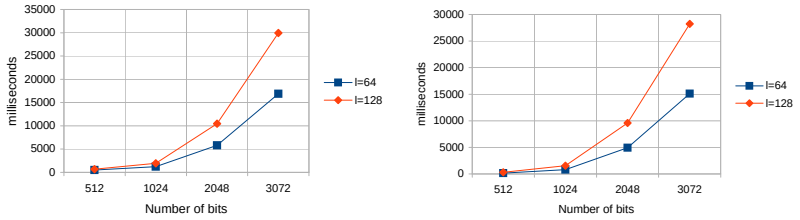


Figure C.4. Execution times (in milliseconds) of the judge in the Identification and Arbitration Protocol versus the number of bits of Paillier's key: (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

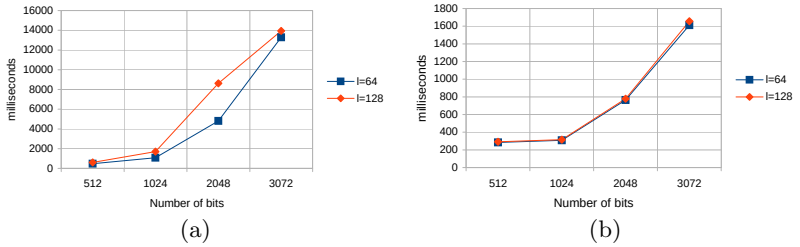


Figure C.5. Execution times (in milliseconds) of the registration/deanonymization authority in the Identification and Arbitration Protocol versus the number of bits of Paillier's key: (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

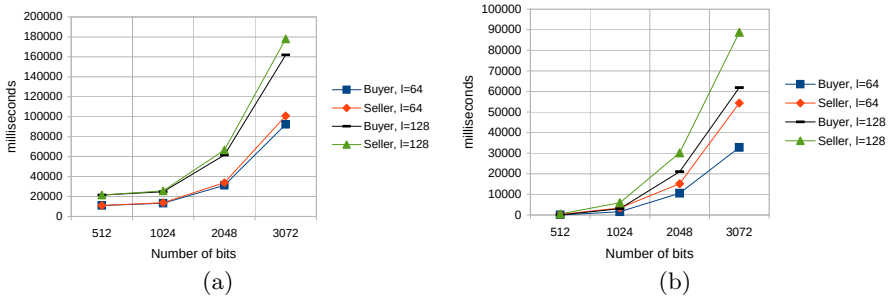


Figure C.6. Execution times (in milliseconds) of the zero knowledge proof π_2 versus the number of bits of Paillier's key: (a) the overall execution time, as the sum of the actual computation time and the handshake latency; (b) the actual computation time

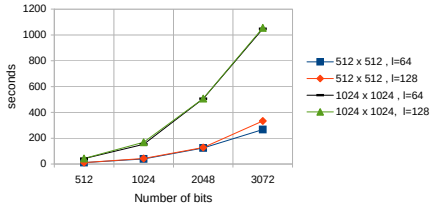


Figure C.7. Execution times (in seconds) of Watermark Embedding versus the number of bits of Paillier's key

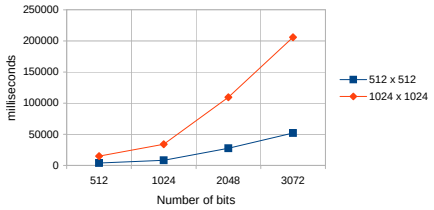


Figure C.8. Execution times (in milliseconds) of Watermarked Image Extraction versus the number of bits of Paillier's key

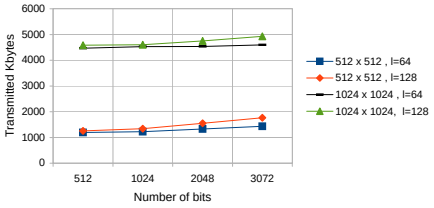


Figure C.9. Exchanged KBytes in the Watermark Generation and Embedding Protocol versus the number of bits of Paillier's key

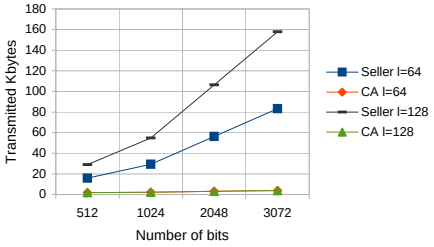


Figure C.10. Exchanged KBytes in the Identification and Arbitration Protocol versus the number of bits of Paillier's key

Bibliography

[1] GNU Multiple Precision Arithmetic Library. Available online: http://gmplib.org/. (p. 207.)

[2] How CPRM and CPPM work from 4C Entity Web site. `http://www.4centity.com/docs/How%20CPRM%20Works.pdf`. (p. 21.)

[3] ISO/IEC 15408 – Common criteria for information technology security evaluation. `http://www.commoncriteriaportal.org/thecc.html`. (pp. 10, 45, 46, and 48.)

[4] Mixmaster. `http://mixmaster.sourceforge.net/`. (p. 76.)

[5] Mixminion. `http://mixminion.net/`. (p. 76.)

[6] NTL: A library for doing number theory. Available online: http://www.shoup.net/ntl/. (p. 207.)

[7] OMA – Open Mobile Alliance: Digital rights management v2.1. `http://www.openmobilealliance.org/technical/release_program/drm_v2_1.aspx`. (p. 23.)

[8] PETs – annual symposium on privacy enhancing technologies homepage. `http://petsymposium.org/`. (p. 75.)

[9] Verance Watermark. `http://www.verance.com/`. (p. 21.)

[10] ISO 17799: Information technology - code of practice for information security management. Technical report, British Standards Institute, 2000. (p. 45.)

[11] Shibboleth overview and requirements. `http://shibboleth.internet2.edu/docs/draft-internet2-shibboleth-requirements-01.html`, February 2001. (p. 137.)

[12] AJAX at the open directory project. `http://dmoz.org/Computers/Programming/Languages/JavaScript/AJAX`, 2006. (p. 139.)

[13] Behealth. `https://www.behealth.be/`, 2006. (p. 139.)

[14] Coral consortium whitepaper. `http://www.coral-interop.org/main/news/Coral.whitepaper.pdf`, February 2006. (p. 19.)

[15] GUIDE project – Creating a European identity manayement architecture for eGovernment. `http://istrg.som.surrey.ac.uk/projects/guide/overview.html`, 2006. (p. 138.)

[16] HIPAA administrative simplification: enforcement; final rule. United States Department of Health & Human service. *Federal Register / Rules and Regulations*, 71(32), 2006. (pp. 51 and 136.)

[17] Liberty Alliance project whitepaper: Personal identity. http://www.projectliberty.org/liberty/content/download/395/2744/file/Personal_Identity.pdf, March 2006. (p. 137.)

[18] Modinis study on identity management in eGovernment: The conceptual framework version 1.1. https://www.cosic.esat.kuleuven.be/modinis-idm/twiki/bin/view.cgi/Main/ConceptualFramework, September 2006. (pp. 134, 137, and 144.)

[19] CBSS – Crossroads bank for social security. http://www.ksz-bcss.fgov.be/En/CBSS.htm, 2007. (p. 138.)

[20] HL7/CDA release 2.0 Clinical Document Architecture. http://www.hl7.org/library/standards_non1.htm, 2007. (p. 140.)

[21] Integrating the Healthcare Enterprise (IHE): IT infrastructure technical framework (revision 4.0). http://www.ihe.net/Technical_Framework/, August 2007. (pp. 139 and 140.)

[22] Windows Cardspace. http://netfx3.com/content/WindowsCardspaceHome.aspx, 2007. (p. 137.)

[23] Custodix home. http://www.custodix.com/, 2008. (p. 139.)

[24] IBBT E-HIP – eHealth Information Platforms. http://www.ibbt.be/en/project/e-hip/, 2008. (pp. 30, 134, and 139.)

[25] IDEM project – Identity Management for eGovernment. https://projects.ibbt.be/idem/, 2008. (pp. 134 and 138.)

[26] Idemix: pseudonymity for e-transactions. http://www.zurich.ibm.com/security/idemix/idemix, 2008. (p. 137.)

[27] Liberty technology glossary working draft. http://xml.coverpages.org/draft-liberty-tech-glossary-08.pdf, 2008. (p. 137.)

[28] PRIME project – Privacy and identity management for Europe. https://www.prime-project.eu/, 2008. (p. 138.)

[29] Privacy guidelines for developing software products and services (version 3.1). Technical report, Microsoft Coorporation, September 2008. (p. 34.)

[30] The security development lifecycle (SDL) version 3.2. Technical report, Microsoft Coorporation, April 2008. http://www.microsoft.com/Downloads/details.aspx?familyid=2412C443-27F6-4AAC-9883-F55BA5B01814&displaylang=en. (p. 34.)

[31] Experimental comparison of attack trees and misuse cases for security threat identification. *Information and Software Technology*, 51(5):916–932, 2009. SPECIAL ISSUE: Model-Driven Development for Secure Information Systems. (p. 41.)

[32] FIDIS project – Future of IDentity in the Information Society. http://www.fidis.net/, 2009. (pp. 31 and 138.)

[33] Integrating the Healthcare Enterprise (IHE) overview. http://www.ihe.net/, 2009. (p. 139.)

[34] PIPEDA – Personal information protection and electronic documents act (2000, c. 5). http://laws.justice.gc.ca/en/showtdm/cs/P-8.6, October 2009. (p. 51.)

[35] SPEED project – Signal Processing in the EncryptEd Domains, 2009. http://www.speedproject.eu/. (p. 31.)

[36] XrML – the digital rights language for trusted content and services. http://www.xrml.org, 2009. (p. 167.)

[37] BD+ technologies LLC. http://www.bdplusllc.com/, 2010. (p. 14.)

[38] Digital content protection LLC. http://www.digital-cp.com/, 2010. (p. 14.)

[39] IBBT Share4Health – Healthcare professional's collaboration space. http://www.ibbt.be/en/project/share4health, 2010. (pp. 30 and 135.)

[40] Martín Abadia and Cédric Fournet. Private authentication. *Theoretical Computer Science*, 322(3):427–476, September 2004. (p. 76.)

[41] Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-lee, Gregory Neven, Pascal, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous ibe, and extensions. In *Advances in Cryptology - CRYPTO'05*, LNCS, pages 205–222. Springer-Verlag, 2005. (p. 77.)

[42] André Adelsbach, Birgit Pfitzmann, and Ahmad-Reza Sadeghi. Proving ownership of digital content. In *Information Hiding*, LNCS, pages 117–133. Springer-Verlag, 1999. (p. 89.)

[43] Andre Adelsbach, Saarbr Ucken, Stefan Katzenbeisser, and Helmut Veith. Watermarking schemes provably secure against copy and ambiguity attacks. In *Proceedings of the Digital Rights Management Workshop*, pages 111–119. ACM, 2003. (p. 173.)

[44] Iskender Agi and Li Gong. An empirical study of secure MPEG video transmission. In *Proc. Symp. on Network and Distributed System Security*, pages 137–144. IEEE Computer Society Press, 1996. (p. 172.)

[45] William Aiello, Steven M. Bellovin, Matt Blaze, Ran Canetti, John Ioannidis, Angelos D. Keromytis, and Omer Reingold. Just fast keying: Key agreement in a hostile internet. *ACM Trans. Inf. Syst. Secur*, 7:2004, 2004. (p. 76.)

[46] Ian Alexander. Misuse cases: Use cases with hostile intent. *IEEE Software*, 20(1):58–66, 2003. (p. 71.)

[47] Anita Allen. Constitutional law and privacy. In Dennis Patterson, editor, *A Companion to Philosophy of Law and Legal Theory*. Oxford University Press, Blackwell, England, 1996. (p. 1.)

[48] Open Mobile Alliance. OMA digital rights management v2.0. http://www.openmobilealliance.org/technical/release_program/drm_v2_0.aspx, July 2008. (p. 172.)

[49] Jonathan Anderson, Claudia Diaz, Joseph Bonneau, and Frank Stajano. Privacy-enabling social networking over untrusted networks. In *Proceedings of the 2nd ACM workshop on Online social networks*, pages 1–6. ACM, 2009. (p. 79.)

[50] Ross Anderson and Fabien Petitcolas. On the limits of steganography. *IEEE Journal of Selected Areas in Communications*, 16:474–481, 1998. (p. 77.)

[51] Guttorm Sindre Andreas and Andreas L. Opdahl. Templates for misuse case description. In *Proceedings of the 7 th International Workshop on Requirements Engineering, Foundation for Software Quality*, pages 4–5, 2001. (pp. 41 and 71.)

[52] Annie I. Antón, Julia Brande Earp, and Angela Reese. Analyzing website privacy requirements using a privacy goal taxonomy. In *RE'02: Proceedings of the 10th Anniversary IEEE Joint International Conference on Requirements Engineering*, pages 23–31. IEEE Computer Society, 2002. (p. 35.)

[53] Apple. How FairPlay works: Apple's iTunes DRM dilemma. `http://www.roughlydrafted.com/RD/RDM.Tech.Q1.07/2A351C60-A4E5-4764-A083-FF8610E66A46.html`, February 2007. (p. 23.)

[54] Claudio A. Ardagna, Jan Camenisch, Markulf Kohlweiss, Ronald Leenes, Gregory Neven, Bart Priem, Pierangela Samarati, Dieter Sommer, and Mario Verdicchio. Exploiting cryptography for privacy-enhanced access control: A result of the PRIME project. *Journal of Computer Security*, 18(1):123–160, January 2010. (p. 77.)

[55] Michael Arnold, Martin Schmucker, and Stephen D. Wolthusen. *Techniques and Applications of Digital Watermarking and Content Protection*. Artech House Publishers Inc., 1 edition, July 2003. (pp. 13, 15, 21, 23, and 169.)

[56] Giuseppe Ateniese, Jan Camenisch, Marc Joye, and Gene Tsudik. A practical and provably secure coalition-resistant group signature scheme. In *Advances in Cryptology - CRYPTO 2000*, LNCS 1880, pages 255–270. Springer-Verlag, 2000. (p. 103.)

[57] Richard Au and Peter Croll. Consumer-centric and privacy-preserving identity management for distributed e-health systems. In *Proceedings of the 41st Hawaii International International Conference on Systems Science*. IEEE Computer Society, January 2008. (p. 137.)

[58] André Bacard. Anonymous.to: Cypherpunk tutorial. `http://www.andrebacard.com/remail.html`. (p. 76.)

[59] Adam Back, Ian Goldberg, and Adam Shostack. Freedom systems 2.1 security issues and analysis. White paper, Zero Knowledge Systems, Inc., May 2001. `http://www.cypherspace.org/adam/pubs/freedom-21.pdf`. (p. 76.)

[60] Mauro Barni and Franco Bartolini. *Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications*. CRC Press, 1 edition, February 2004. (pp. 15, 103, and 169.)

[61] Mark Barry. Cryptography in home entertainment a look at content scrambling in DVDs. `http://www.math.ucsd.edu/~crypto/Projects/MarkBarry/index.htm`, June 2004. (p. 21.)

[62] Ann Bartow. A feeling of unease about privacy law. *University of Pennsylvania Law Review*, 154:52–62, 2006. (p. 9.)

[63] Filipe Beato, Markulf Kohlweiss, and Karel Wouters. Enforcing access control in social networks. HotPets, 2009. `http://www.cosic.esat.kuleuven.be/publications/article-1240.pdf`. (p. 79.)

[64] Mihir Bellare and Oded Goldreich. On defining proofs of knowledge. In Ernest F. Brickell, editor, *Advances in Cryptology - CRYPTO'92*, LNCS 740, pages 390–420. Springer-Verlag, 1992. (p. 102.)

[65] Mihir Bellare, Daniele Micciancio, and Bogdan Warinschi. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In *Topics in Cryptology - Eurocrypt'03*, LNCS 2656, pages 614–629. Springer-Verlag, 2003. (p. 77.)

[66] Mihir Bellare, Haixia Shi, and Chong Zhang. Foundations of group signatures: the case of dynamic groups. In *Topics in Cryptology - CT-RSA'05*, LNCS 3376, pages 136–153. Springer-Verlag, 2005. (pp. 77, 100, 101, 109, 198, 199, 201, and 205.)

[67] Oliver Berthold, Hannes Federrath, and Stefan Köpsell. Web MIXes: A system for anonymous and unobservable Internet access. In H. Federrath, editor, *Proceedings of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability*, LNCS 2009, pages 115–129. Springer-Verlag, July 2000. (p. 76.)

[68] Ingrid Biehl and Bernd Meyer. Protocols for collusion-secure asymmetric fingerprinting. In *Proc. 14th STACS*, LNCS 1200, pages 213–222. Springer-Verlag, 1997. (p. 88.)

[69] G. R. Blakley, Catherine Meadows, and George B. Purdy. Fingerprinting long forgiving messages. In *Advances in Cryptology - CRYPTO'85*, LNCS, pages 180–189. Springer-Verlag, 1985. (p. 87.)

[70] John Blau. Cracks appear in Bluetooth security. `http://www.networkworld.com/news/2004/0211cracksappear.html`, Feb. 2004. (p. 168.)

[71] Bernd Blobel. Authorisation and access control for electronic health record systems. *International Journal of Medical Informatics*, 73(3):251–257, 2004. (p. 19.)

[72] Jeffrey A. Bloom, Ingemar J. Cox, Senior Member, Ton Kalker, Jean-Paul M. G. Linnartz, Matthew L. Miller, and C. Brendan S. Traw. Copy protection for DVD video. In *Proceedings of the IEEE*, pages 1267–1276, 1999. (p. 21.)

[73] Jens-Matthias Bohli and Andreas Pashalidis. Relations among privacy notions. In *Financial Cryptography*, LNCS, pages 362–380. Springer-Verlag, 2009. (pp. 11, 12, and 26.)

[74] Dan Boneh and James Shaw. Collusion-secure fingerprinting for digital data (extended abstract). In *Advances in Cryptology - CRYPTO'95*, LNCS. Springer-Verlag. (p. 87.)

[75] Nikita Borisov, Ian Goldberg, and Eric Brewer. Off-the-record communication, or, why not to use pgp. In *Proceedings of the 2004 ACM workshop on Privacy in the electronic society*, pages 77–84. ACM, 2004. (p. 76.)

[76] Stefan Brands and David Chaum. Distance-bounding protocols (extended abstract). In *Advances in Cryptology - EUROCRYPT'93*, LNCS 765, pages 344–359. Springer-Verlag, 1993. (p. 76.)

[77] Jack Brassil, Steven H. Low, Nicholas F. Maxemchuk, and Lawrence O'Gorman. Electronic marking and identification techniques to discourage document copying. *IEEE Journal on Selected Areas in Communications*, 13(8):1495–1504, 1995. (p. 87.)

[78] Travis Breaux and Annie Antón. Analyzing regulatory rules for privacy and security requirements. *IEEE Trans. Softw. Eng.*, 34(1):5–20, 2008. (p. 51.)

[79] Travis D. Breaux, Annie I. Anton, Kent Boucher, and Merlin Dorfman. Legal requirements, compliance and practice: An industry case study in accessibility. In *RE'08: Proceedings of the 16th IEEE International Requirements Engineering Conference (RE'08)*, pages 43–52. IEEE Society Press, September 2008. (p. 51.)

[80] Ernest F. Brickell, Jan Camenisch, and Liqun Chen. Direct anonymous attestation. In *ACM Conference on Computer and Communications Security*, pages 132–145. ACM, 2004. (p. 25.)

[81] David Brin. *The Transparent Society: Will Technology Force Us to Choose Between Privacy and Freedom?* Basic Books, June 1999. (p. 2.)

[82] Christian Cachin. On the foundations of oblivious transfer. In *Advances in Cryptology – Eurocrypt '98*, LNCS 1403, pages 361–374. Springer-Verlag, 1998. (p. 77.)

[83] Jan Camenisch. Efficient anonymous fingerprinting with group signatures. In *Advances in Cryptology - ASIACRYPT 2000*, LNCS, pages 415–428. Springer-Verlag, 2000. (pp. 87 and 89.)

[84] Jan Camenisch and Ivan Damgård. Verifiable encryption, group encryption, and their applications to separable group signatures and signature sharing schemes. In *Advances in Cryptology - ASIACRYPT 2000*, LNCS, pages 331–345. Springer-Verlag, 2000. (p. 77.)

[85] Jan Camenisch and Els Van Herreweghen. Design and implementation of the idemix anonymous credential system. In *Proceedings of the 9th ACM conference on Computer and communications security*, pages 21–30. ACM, 2002. (p. 137.)

[86] Jan Camenisch and Anna Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *Advances in Cryptology - CRYPTO'04*, LNCS 3152, pages 56–72. Springer-Verlag, 2004. (p. 76.)

[87] Jan Camenisch, Ueli M. Maurer, and Markus Stadler. Digital payment systems with passive anonymity-revoking trustees. In *ESORICS*, LNCS, pages 33–43. Springer-Verlag, 1996. (p. 103.)

[88] Jan Camenisch and Victor Shoup. Practical verifiable encryption and decryption of discrete logarithms. In *Advances in Cryptology - CRYPTO 2003*, LNCS 2729, pages 126–144. Springer-Verlag, 2003. (pp. 103, 110, and 119.)

[89] Jan Camenisch and Markus Stadler. Proof systems for general statements about discrete logarithms. Technical Report TR 260, Institute for Theoretical Computer Science, ETH Zürich, 1997. (p. 103.)

[90] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Annual IEEE Symposium on Foundations of Computer Science – FOCS*, pages 136–145, 2001. (pp. 95 and 105.)

[91] Barbara Carminati and Elena Ferrari. Privacy-aware collaborative access control in web-based social networks. In *Proc. of the 22nd IFIP WG 11.3 Working Conference on Data and Applications Security (DBSEC2008)*, pages 81–96. Springer Berlin / Heidelberg, July 2008. (pp. 77, 80, and 81.)

[92] Fred H. Cate. *Privacy in the Information Age*. Brookings Institution Press, November 1997. (p. 4.)

[93] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, 1981. (p. 76.)

[94] David Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985. (p. 76.)

[95] David Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1(1):65–75, 1988. (p. 76.)

[96] David Chaum and Eugene van Heyst. Group signatures. In *Advances in Cryptology - EUROCRYPT'91*, LNCS 547, pages 257–265. Springer-Verlag, 1991. (p. 100.)

[97] Brian Chen and Gregory W. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, 47(4):1423–1443, 2001. (p. 209.)

[98] Benny Chor, Oded Goldreich, Eyal Kushilevitz, and Madhu Sudan. Private information retrieval. In *Journal of the ACM*, pages 41–50, 1998. (p. 77.)

[99] Sebastian Clauß, Andreas Pfitzmann, Marit Hansen, and Els Van Herreweghen. Privacy-enhancing identity management. The IPTS Report 67, pages 8-16, September 2002. (pp. 77 and 81.)

[100] Julie E. Cohen. DRM and privacy. *Berkeley Technological Law Journal*, 18:575–617, 2003. `http://ssrn.com/abstract=372741`. (p. 22.)

[101] Ingemar Cox, Joe Kilian, Tom Leighton, and Talal Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Trans. on Image Processing*, 6(12):1673–1687, 1997. (p. 110.)

[102] Ingemar Cox, Matthew Miller, Jeffrey Bloom, and Mathew Miller. *Digital Watermarking: Principles & Practice*. The Morgan Kaufmann Series in Multimedia Information and Systems. Morgan Kaufmann, 2001. (pp. 15 and 169.)

[103] Ronald Cramer, Ivan Damgard, and Philip Mackenzie. Efficient zero-knowledge proofs of knowledge without intractability assumptions. In Hideki Imai and Yuliang Zheng, editors, *International Workshop on Theory and Practice in Public Key Cryptography – PKC*, LNCS 1751, pages 354–372. Springer-Verlag, 2000. (p. 128.)

[104] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. A secure and optimally efficient multi-authority election scheme. In *Advances in Cryptology – EUROCRYPT 1997*, LNCS, pages 103–118. Springer-Verlag, 1997. (p. 109.)

[105] Scott Craver, Nasir Memon, Boon-Lock Yeo, and Minerva M. Yeung. Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks, and implications. *IEEE Journal on Selected Areas in Communications*, 16(4):573–586, 1998. (pp. 95 and 118.)

[106] CSS. Content scramble system (CSS) official website. `http://www.dvdcca.org/css/`. (p. 21.)

[107] Joan Daemen and Vincent Rijmen. AES proposal: Rijndael, 1998. (p. 147.)

[108] Ivan Damgård and Mads Jurik. A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In *4th International Workshop on Practice and Theory in Public-Key Cryptography*, LNCS 1992, pages 119–136. Springer-Verlag, 2001. (pp. 77, 101, 102, 109, 119, and 128.)

[109] George Danezis. Talk: Introduction to privacy technology. `http://research.microsoft.com/en-us/um/people/gdane/talks/Privacy_Technology_cosic.pdf`, 2007. (pp. 44, 83, and 176.)

[110] George Danezis. Privacy as security. `http://www.secappdev.org/handouts/2008/privacy.pdf`, 2008. (p. 44.)

[111] George Danezis. Talk: an introduction to U-Prove privacy protection technology, and its role in the Identity Metasystem – What future for privacy technology. `http://www.petsfinebalance.com/agenda/index.php`, 2008. (pp. 44, 83, and 176.)

[112] George Danezis, Claudia Diaz, and Paul Syverson. *Systems for Anonymous Communication, in CRC Handbook of Financial Cryptography and Security*, page 61. Chapman & Hall, 2009. (pp. 59 and 61.)

[113] Judith DeCew. *Privacy – from Stanford Encyclopedia of Philosophy (First published Tue May 14, 2002; substantive revision Mon Sep 18, 2006)*, 2006. `http://plato.stanford.edu/entries/privacy/`. (p. 7.)

[114] DELL. *Blu-ray Disc next-generation optical storage: protecting content on the BD-ROM (white paper)*, October 2006. `http://www.dell.com/downloads/global/vectors/brcp.pdf`. (pp. 15 and 21.)

[115] Mina Deng, Tiziano Bianchi, Alessandro Piva, and Bart Preneel. An efficient buyer-seller watermarking protocol based on composite signal representation. In *Proceedings of the 11th ACM workshop on Multimedia and security*, pages 9–18, Princeton, New Jersey, USA, 2009. ACM New York, NY, USA. (pp. 28, 29, 77, 95, 109, 178, 207, 208, and 209.)

[116] Mina Deng, Tiziano Bianchi, Alessandro Piva, and Bart Preneel. Efficient implementation of a buyer-seller watermarking protocol using a composite signal representation. In J. Guajardo and A. Piva, editors, *The International Workshop on Signal Processing in the EncryptEd Domain (SPEED 2009)*, pages 22–41, Lausanne Switzerland, 2009. (pp. 28, 29, 95, 178, 207, 208, and 209.)

[117] Mina Deng, Tiziano Bianchi, Alessandro Piva, Alfredo Rial, and Bart Preneel. Anonymous buyer-seller watermarking protocols – efficient implementations. Technical report, K.U.Leuven & Università di Firenze, April 2010. (pp. 29, 95, 178, and 207.)

[118] Mina Deng and Danny De Cock. Towards cross-context identity management framework in e-health. In *Managing Identity in New Zealand – Identity Conference*, pages 1–10, Wellington, New Zealand, April 2008. (pp. 28 and 135.)

[119] Mina Deng, Danny De Cock, and Bart Preneel. An interoperable cross-context architecture to manage distributed personal e-health information. In M. M. Cunha, R. Simoes, and A. Tavares, editors, *Handbook of Research on Developments in e-Health and Telemedicine: Technological and Social Perspectives, ISBN: 978-1-61520-670-4*, chapter 27, pages 576–602. Hershey, PA, USA: IGI Global, Inc., 2009. (pp. 28 and 135.)

[120] Mina Deng, Danny De Cock, and Bart Preneel. Towards a cross-context identity management framework in e-health. *Online Information Review*, 33(3):422–442, 2009. (pp. 28 and 135.)

[121] Mina Deng, Lothar Fritsch, and Klaus Kursawe. Personal rights management-enabling privacy rights in digital online content. In Jana Dittmann, Stefan Katzenbeisser, and Andreas Uhl, editors, *Communications and Multimedia Security*, LNCS 3677, pages 266–268. Springer, 2005. (pp. 28 and 162.)

[122] Mina Deng, Lothar Fritsch, and Klaus Kursawe. Personal rights management - taming camera-phones for individual privacy enforcement. In George Danezis and Philippe Golle, editors, *Privacy Enhancing Technologies*, LNCS 4258, pages 172–189. Springer, 2006. (pp. 28 and 162.)

[123] Mina Deng and Bart Preneel. Attacks on two buyer-seller watermarking protocols and an improvement for revocable anonymity. In Fei Yu, Qi Luo, Yongjun Chen, and Zhigang Chen, editors, *International Symposium on Electronic Commerce and Security*, pages 923–929. IEEE Computer Society, 2008. (pp. 28 and 95.)

[124] Mina Deng and Bart Preneel. Attacks on two buyer-seller watermarking protocols and an improvement for revocable anonymity. *International Journal of Intelligent Information Technology Application*, 1(2):53–64, 2008. (pp. 28 and 95.)

[125] Mina Deng and Bart Preneel. On secure and anonymous buyer-seller watermarking protocol. In Abdelhamid Mellouk, Jun Bi, Guadalupe Ortiz, Dickson K. W. Chiu, and Manuela Popescu, editors, *International Conference on Internet and Web Applications and Services*, pages 524–529. IEEE Computer Society, 2008. (pp. 28 and 95.)

[126] Mina Deng and Bart Preneel. On secure buyer-seller watermarking protocols with revocable anonymity. In Kyeong Kang, editor, *E-Commerce, ISBN: 978-953-7619-98-5*, chapter 11, pages 184–202. IN-TECH Education and Publishing, Vienna, Austria, 2009. (pp. 28 and 95.)

[127] Mina Deng and Bart Preneel. Privacy and data protection architecturen, deliverable 1.2.2: Share4Health – healthcare professional's collaboration space. Technical report, IBBT-COSIC, K.U.Leuven, Nov. 2009. (p. 183.)

[128] Mina Deng, Riccardo Scandariato, Danny De Cock, Bart Preneel, and Wouter Joosen. Identity in federated electronic healthcare. In *1st IFIP Wireless Days (WD'08) – the 6th IFIP Network Control conference (Netcon'08)*, pages 1–5, Dubai, UAE, Nov. 2008. IEEE. (pp. 28 and 135.)

[129] Mina Deng, Li Weng, and Bart Preneel. Anonymous buyer-seller watermarking protocol wth additive homomorphism. In Pedro A. Amado Assunção and Sérgio M. M. de Faria, editors, *International Conference on Signal Processing and Multimedia Applications*, pages 300–307. INSTICC Press, 2008. (pp. 28, 29, and 95.)

[130] Mina Deng, Kim Wuyts, Riccardo Scandariato, Bart Preneel, and Wouter Joosen. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirement Engineering Journal special issue on Data Privacy, To appear*, page 27, 2010. (pp. 28 and 36.)

[131] Digital Content Protection LLC. *High-bandwidth Digital Content Protection System – Interface Independent Adaptation (Revision 2.0)*, October 2008. `http://www.digital-cp.com/files/static_page_files/2C1C0F30-0E09-E813-BFAB6BAAE8A76080/HDCP%20Interface%20Independent%20Adaptation%20Specification%20Rev2_0.pdf`. (pp. 14 and 21.)

[132] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *In Proceedings of the 13th USENIX Security Symposium*, pages 303–320, Aug. 2004. (pp. 76, 80, 81, 82, 110, and 120.)

[133] Alexander Dix. Das Recht am eigenen Bild – Anachronismus im Zeitalter des Internet? In *Mediale (Selbst-)Darstellung und Datenschutz, Konferenz des LfD NRW*, 2000. (p. 159.)

[134] EBU. *Functional model of a conditional access system, EBU Technical Review*. EBU Project Group B/CA, October 1995. `http://www.ebu.ch/en/technical/trev/trev_266-ca.pdf`. (p. 19.)

[135] EC. European convention on human rights. *Martinus Nijhoff Publishers*, 1987. (p. 136.)

[136] J. Eggers, J. Su, and B. Girod. A blind watermarking scheme based on structured codebooks. In *Secure Images and Image Authentication, IEE Colloq.*, pages 4/1–4/6, Apr 2000. (p. 99.)

[137] Ahmed M. Eskicioglu and Edward J. Delp. Protection of multimedia content in distribution networks. In Borko Furht and Darko Kirovski, editors, *Multimedia Security Handbook*, chapter 1, pages 3–62. CRC Press, 2004. (pp. 13 and 16.)

[138] Ahmet M. Eskicioglu. A key transport protocol based on secret sharing – an application to message authentication. In *Communications and Multimedia Security*, LNCS. Springer-Verlag, 2001. (p. 19.)

[139] Ahmet M. Eskicioglu and Edward J. Delp. Overview of multimedia content protection in consumer electronics devices. *Signal Processing: Image Communication*, 16(7):681–699, 2001. (p. 16.)

[140] EU. Directive 95/46/EC of the European parliament and of the council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. *Official Journal of the European Communities*, 281:31–50, 1995. `http://ec.europa.eu/justice_home/fsj/privacy/index_en.htm`. (pp. 8, 50, 51, 136, 159, 161, and 167.)

[141] Independent Security Evaluators. *Content Protection for Optical Media – A Comparison of Self-Protecting Digital Content and AACS*, May 2005. `http://securityevaluators.com/pdf/spdc_aacs_2005.pdf`. (p. 14.)

[142] Mark Evered and Serge Bögeholz. A case study in access control requirements for a health information system. In *ACSW Frontiers*, pages 53–61, 2004. (p. 19.)

[143] Joan Feigenbaum, Michael J. Freedman, Tomas Sander, and Adam Shostack. Privacy engineering for digital rights management systems. In *Digital Rights Management Workshop*, pages 76–105, 2001. (p. 22.)

[144] Yair Frankel, Yiannis Tsiounis, and Moti Yung. Indirect disclosure proof: Achieving. efficient fair off-line e-cash. In *Advances in Cryptology - ASIACRYPT '96*, LNCS 1163, pages 286–300. Springer-Verlag, 1996. (p. 103.)

[145] Franco Frattolillo. Watermarking protocol for web context. *IEEE Trans. on Information Forensics and Security*, 2(3):350–363, Sept. 2007. (p. 95.)

[146] Jiri Fridrich and Miroslav Goljan. Robust hash functions for digital watermarking. In *Proceedings of the International Conference on Information Technology: Coding and Computing*, pages 173–178, March 2000. (p. 171.)

[147] David D. Friedman. Privacy and technology. In Ellen Frankel Paul, Jr. Fred D. Miller, and Jeffrey Paul, editors, *The Right to Privacy*, pages 186–212. Cambridge University Press, 2000. (p. 2.)

[148] Borko Furht and Darko Kirovski. *Multimedia Security Handbook*. CRC Press, 1 edition, December 2004. (pp. xxvii, 15, 16, 17, 18, 19, 20, and 21.)

[149] Taher El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Advances in Cryptology - CRYPTO '84*, LNCS 196, pages 10–18. Springer-Verlag, 1985. (pp. 101 and 109.)

[150] Ruth Gavison. Privacy and the limits of law. *Yale Law Journal*, 89:421–471, 1980. (p. 7.)

[151] Christos K. Georgiadis, Ioannis Mavridis, George Pangalos, and Roshan K. Thomas. Flexible team-based access control using contexts. In *ACM symposium on Access control models and technologies*, pages 21–27. ACM, 2001. (p. 77.)

[152] Bok-Min Goi, Raphael Chung-Wei Phan, Yanjiang Yang, Feng Bao, Robert H. Deng, and M. U. Siddiqi. Cryptanalysis of two anonymous buyer-seller watermarking protocols and an improvement for true anonymity. In *Applied Cryptography and Network Security*, LNCS 2587, pages 369–382. Springer-Verlag, 2004. (pp. 90, 91, and 92.)

[153] Bettina Goldmann. Copy protection by DRM in the EU and germany: Legal aspects. In *Digital Rights Management - Technological, Economic, Legal and Political Aspects*, pages 502–519. Springer, 2003. (p. 21.)

[154] David M. Goldschlag, Michael G. Reed, and Paul F. Syverson. Hiding Routing Information. In R. Anderson, editor, *Proceedings of Information Hiding: First International Workshop*, pages 137–150. Springer-Verlag, LNCS 1174, May 1996. (p. 76.)

[155] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984. (pp. 101 and 200.)

[156] Calvin C Gotlieb. Privacy: A concept whose time has come and gone. In D. Lyon and E. Zureik, editors, *Surveillance, Computers and Privacy*, chapter 8, pages 156–171. University of Minnesota Press, 1996. (p. 4.)

[157] Derrick Grover. *The Protection of Computer Software – its Technology and Applications (British Computer Society Monographs in Informatics)*. Cambridge University press, 1 edition, March 1989. (p. 87.)

[158] Marit Hansen, Peter Berlich, Jan Camenisch, Sebastian Clauß, Andreas Pfitzmann, and Michael Waidner. Privacy-enhancing identity management. *Information Security Technical Report (ISTR)*, 9(1):35–44, 2004. `http://dx.doi.org/10.1016/S1363-4127(04)00014-7`. (pp. 77 and 81.)

[159] Frank Hartung and Friedhelm Ramme. Digital rights management and watermarking for multimedia content for M-commerce. *IEEE Communications Magazine*, 38(11):78–84, November 2000. (p. 172.)

[160] Heather Havenstein. New facebook ad techniques raise privacy concerns. `http://www.pcworld.com/article/139494/new_facebook_ad_techniques_raise_privacy_concerns.html`, 2009. (p. 3.)

[161] Alejandro Hevia and Daniele Micciancio. An indistinguishability-based characterization of anonymous channels. In *Privacy Enhancing Technologies*, pages 24–43, 2008. (pp. 10, 11, 12, and 26.)

[162] Higgins. Eclipse Higgins Projec. `http://eclipse.org/higgins/`. (p. 137.)

[163] Michael Howard and Steve Lipner. *The Security Development Lifecycle*. Microsoft Press, 2006. (pp. 33, 38, and 68.)

[164] Luigi Lo Iacono. Multi-centric universal pseudonymisation for secondary use of the EHR. In *From Genes to Personalized HealthCare: Grid Solutions for the Life Sciences – Proceedings of HealthGrid 2007*, volume 126, pages 239–247, 2007. (p. 137.)

[165] IBM. Enterprise privacy authorization language: EPAL 1.2. `http://www.w3.org/Submission/2003/SUBM-EPAL-20031110/`. (p. 78.)

[166] Ibrahim M. Ibrahim, Sherif Hazem Nour El-Din, and Abdel Fatah A. Hegazy. An effective and secure buyer-seller watermarking protocol. In *International Symposium on Information Assurance and Security (IAS 2007)*, pages 21–28. IEEE Computer Society, Aug. 2007. (pp. 90, 91, 92, 96, and 98.)

[167] Digimarc MarcSpider image tracking. Digimarc and cobion announce new partnership to provide enhanced tracking of watermarked images online. `https://www.digimarc.com/media/release.asp?newsID=63`, 2001. (p. 170.)

[168] ODRL Initiative. The open digital rights language (ODRL) initiative. `http://www.odrl.net`, 2009. (p. 167.)

[169] CMU Software Engineering Institute. OCTAVE – (Operationally Critical Threat, Asset, and Vulnerability Evaluation). `http://www.cert.org/octave/`. (p. 70.)

[170] Ji-Hwan Park Jae-Gwi Choi, Kouichi Sakurai. Does it need trusted third party? design of buyer-seller watermarking protocol without trusted third party. In *Applied Cryptography and Network Security*, LNCS 2846, pages 265–279. Springer-Verlag, 2003. (pp. 90, 91, and 92.)

[171] Mark Johnson and Kannan Ramchandran. Dither-based secure image hashing using distributed coding. In *Proceedings of the International Conference on Image Processing*, pages 751–754, 2003. (p. 171.)

[172] Hak-Soo Ju, Hyung-Jeong Kim, Dong-Hoon Lee, and Jong-In Lim. An anonymous buyer-seller watermarking protocol with anonymity control. *Information Security and Cryptology – ICISC 2002*, pages 421–432, Nov. 2002. (pp. 90, 91, and 92.)

[173] Christos Kalloniatis, Evangelia Kavakli, and Stefanos Gritzalis. Addressing privacy requirements in system design: the pris method. *Requirements Engineering*, 13(3):241–255, September 2008. `http://dx.doi.org/10.1007/s00766-008-0067-3`. (pp. 35 and 75.)

[174] Stefan Katzenbeisser and Fabien A.P. Petitcolas. *Information hiding techniques for steganography and digital watermarking*. Artech Print on Demand, 2000. (p. 169.)

[175] Auguste Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, IX:5–83, Jan. 1883. (p. 99.)

[176] Donald Kerr. *Remarks and Q&A by the Principal Deputy Director of National Intelligence Dr. Donald Kerr*. GEOINT Symposium Sponsored by the United States Geospatial Intelligence Foundation, October 2007. `http://www.dni.gov/speeches/20071023_speech.pdf`. (pp. 3 and 6.)

[177] Ian Kerr and Steve Mann. Over and under the valences of veillance – exploring equiveillance. blog*on*nymity, January 2006. (p. 2.)

[178] Aggelos Kiayias and Moti Yung. The vector-ballot e-voting approach. In *Financial Cryptography*, LNCS 3110, pages 72–89. Springer-Verlag, 2004. (p. 109.)

[179] Joe Kilian and Erez Petrank. Identity escrow. In *Advances in Cryptology - CRYPTO '98*, LNCS 1462, pages 169–185. Springer-Verlag, 1998. (p. 103.)

[180] Larry Korba and Steve Kenny. Towards meeting the privacy challenge: Adapting DRM. In *Digital Rights Management Workshop*, LNCS 2696, pages 118–136. Springer-Verlag, 2002. (p. 167.)

[181] Paul Koster. Person-based and domain-based digital rights management. In *Security, Privacy, and Trust in Modern Data Management*, chapter 20, pages 303–316. Springer Berlin Heidelberg, June 2007. (p. 23.)

[182] Paul Koster, Frank Kamperman, Peter Lenoir, and Koen Vrielink. Identity-based DRM: Personal entertainment domain. In *Communications and Multimedia Security*, LNCS, pages 42–54. Springer-Verlag, 2005. (p. 23.)

[183] Suleyman Serdar Kozat, Ramarathnam Venkatesan, and Mehmet Kivanç Mihçak. Robust perceptual image hashing via matrix invariants. In *IEEE Int. Conf. Image Process – ICIP*, pages 3443–3446, 2004. (p. 171.)

[184] Steve Kremer. *Formal Analysis of Optimistic Fair Exchange Protocols*. PhD thesis, Université Libre de Bruxelles, 2004. (p. 184.)

[185] William Ku and Chi-Hung Chi. Survey on the technological aspects of digital rights management. In *Information Security*, LNCS 3225, pages 391–403. Springer-Verlag, 2004. (p. 172.)

[186] Minoru Kuribayashi and Hatsukazu Tanaka. Fingerprinting protocol for images based on additive homomorphic property. *IEEE Trans. on Image Processing*, 14(12):2129–2139, Dec. 2005. (p. 123.)

[187] Martin Kutter and Fabien A. P. Petitcolas. A fair benchmark for image watermarking systems. In *SPIE Security and Watermarking of Multimedia Contents*, volume 3657, pages 226–239, 1999. (p. 99.)

[188] Scott Lederer, Jason I. Hong, Anind K. Dey, and James A. Landay. Personal privacy through understanding and action: Five pitfalls for designers. *Personal and Ubiquitous Computing*, 8:440–454, 2004. (pp. 49, 78, and 83.)

[189] Chin-Laung Lei, Pei-Ling Yu, Pan-Lung Tsai, and Ming-Hwa Chan. An efficient and anonymous buyer-seller watermarking protocol. *IEEE Trans. on Image Processing*, 13(12):1618–1626, 2004. (pp. 90, 91, 92, 96, and 116.)

[190] Hong Li and Milan Petković. Drm for protecting personal content. In *Security, Privacy, and Trust in Modern Data Management*, chapter 22, pages 333–346. Springer Berlin Heidelberg, June 2007. (p. 25.)

[191] MSDN Library. Improving web application security: Threats and countermeasures. (p. 70.)

[192] Heather Richter Lipford, Andrew Besmer, and Jason Watson. Understanding privacy settings in facebook with an audience view. In Elizabeth F. Churchill and Rachna Dhamija, editors, *Proceedings of the 1st Conference on Usability, Psychology, and Security*. USENIX Association Berkeley, CA, USA, 2008. `http://www.usenix.org/events/upsec08/tech/full_papers/lipford/lipford.pdf`. (pp. 78 and 83.)

[193] K. J. Ray Liu, Wade Trappe, Jane Z. Wang, Min Wu, and Hong Zhao. *Multimedia Fingerprinting Forensics for Traitor Tracing*. EURASIP Book Series on Signal Processing and Communications. Hindawi Publishing Co., 2005. (p. 87.)

[194] Lin Liu, Eric Yu, and John Mylopoulos. Security and privacy requirements analysis within a social setting. In *Proceeding of the IEEE International Conference on Requirements Engineering*, pages 151–161. IEEE Computer Society, 2003. (p. 35.)

[195] Qiong Liu, Reihaneh Safavi-Naini, and Nicholas Paul Sheppard. Digital rights management for content distribution. In *ACSW Frontiers*, pages 49–58, 2003. (pp. xxvii and 17.)

[196] Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *International Conference on Data Engineering (ICDE'06)*, page 24. IEEE Computer Society, 2006. (p. 77.)

[197] Benoit M. Macq, Jana Dittmann, and Edward J. Delp. Benchmarking of image watermarking algorithms for digital rights management. *Proceedings of the IEEE*, 92(6):971–984, 2004. (pp. 170, 172, and 173.)

[198] B.M. Macq and J. J. Quisquater. Cryptology for digital TV broadcasting. *Proceedings of the IEEE*, 83(6):944–957, June 1995. (p. 19.)

[199] Maurice Maes, Ton Kalker, Jean paul Linnartz, Joop Talstra, Geert Depovere, and Jaap Haitsma. Digital watermarking for DVD video copy protection: What issues play a role in designing an effective system? *IEEE Signal Processing Magazine*, 17:47–57, 2000. (p. 21.)

[200] Wenbo Mao and Kenneth G. Paterson. *On the Plausible Deniability Feature of Internet Protocols*, 2002. `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.5.2401`. (p. 48.)

[201] Erika McCallister, Tim Grance, and Karen Kent. Guide to protecting the confidentiality of personally identifiable information (PII) (draft). Technical report, National Institute of Standards and Technology (U.S.), 2009. (p. 49.)

[202] Gary McGraw. *Software Security: Building Security In*. Addison-Wesley Professional, 2006. (p. 33.)

[203] Nasir D. Memon and Ping Wah Wong. A buyer-seller watermarking protocol. *IEEE Trans. on Image Proc.*, 10(4):643–649, 2001. (pp. 90, 91, 92, 93, and 116.)

[204] Alfred J. Menezes, Paul C. Van Oorschot, Scott A. Vanstone, and R. L. Rivest. *Handbook of Applied Cryptography*. CRC Press, December 1996. `http://www.cacr.math.uwaterloo.ca/hac/`. (pp. 15 and 77.)

[205] Mehmet Kivanç Mihçak and Ramarathnam Venkatesan. New iterative geometric methods for robust perceptual image hashing. In *ACM CCS-8 Workshop on Security and Privacy in Digital Rights Management*, pages 13–21. Springer-Verlag, 2002. (p. 171.)

[206] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, Oct. 2005. (p. 171.)

[207] Seiya Miyazaki, Nancy Mead, and Justin Zhan. Computer-aided privacy requirements elicitation technique. *Asia-Pacific Conference on Services Computing. 2006 IEEE*, pages 367–372, 2008. (p. 35.)

[208] Vishal Monga and Brian L. Evans. Robust perceptual image hashing using feature points. In *Proc. IEEE Int. Conf. on Image Processing*, pages 677–680, Oct. 2004. (p. 171.)

[209] Vishal Monga, Divyanshu Vats, and Brian L. Evans. Image authentication under geometric attacks via structure matching. In *Proc. IEEE Int. Conf. on Multimedia & Expo*, pages 229–232. Amsterdam, The Netherlands, July 2005. (p. 171.)

[210] Adam Moore. Intangible property: Privacy, power, and information control. *American Philosophical Quarterly*, 35:365–378, 1998. (p. 7.)

[211] Ryoichi Mori and Masaji Kawahara. Superdistribution: The concept and the architecture. *IEICE Transactions (1976-1990)*, E73-E(7):1133–1146, July 1990. (p. 17.)

[212] Ira Moskowitz, Richard E. Newman, Daniel P. Crepeau, and Allen R. Miller. Covert channels and anonymizing networks. In *In Workshop on Privacy in the Electronic Society*, pages 79–88. ACM, 2003. (p. 77.)

[213] John Mylopoulos, Lawrence Chung, and Brian Nixon. Representing and using non-functional requirements: A process-oriented approach. *IEEE Transactions on Software Engineering*, 18:483–497, 1992. (p. 34.)

[214] Moni Naor. Deniable ring authentication. In *Advances in Cryptology - CRYPTO 2002*, LNCS 2442, pages 481–498. Springer-Verlag, 2002. (p. 76.)

[215] Moni Naor and Kobbi Nissim. Communication complexity and secure function evaluation. *CoRR*, cs.CR/0109011, 2001. (p. 76.)

[216] Australia's national privacy regulator. Australia's national privacy regulator: Privacy act. `http://www.privacy.gov.au/law/act`. (p. 51.)

[217] Deborah Nelson. *Pursuing Privacy in Cold War America*. Columbia University Press, January 2002. (p. 3.)

[218] ABC News. Students protest after teacher suspended for Bush-Hitler comments. `http://abcnews.go.com/GMA/story?id=1679439`, March 2006. (p. 185.)

[219] NHS. British national health service NHS choices: The website for England's NHS. `http://www.nhs.uk/`, 2009. (p. 140.)

[220] Nikos Nikolaidis, Sofia Tsekeridou, Anastasios Tefas, Vassilios Solachidis, Athanasios Nikolaidis, and Ioannis Pitas. A benchmarking protocol for watermarking methods. In *Int. Conf. on Image Processing (ICIP 01)*, pages 1023–1026, 2001. (p. 170.)

[221] NIST. Risk management guide for information technology systems, special publication 800-30. http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf. (p. 70.)

[222] OASIS. eXtensible access control markup language: XACML 3.0. http://xml.coverpages.org/xacml.html. (p. 78.)

[223] OECD. Guidelines on the protection of privacy and transborder flows of personal data, organization for economic cooperation and development. http://www.oecd.org/document/18/0,2340,en_2649_34255_1815186_1_1_1_1,00.html, 1980. (p. 51.)

[224] US Department of Justice. Workforce Investment Act of 1998, SEC. 508. electronic and information technology. http://www.justice.gov/crt/508/508law.php. (p. 51.)

[225] ECRYPT workshop One DRM to Rule them All. Ton kalker. https://www.cosic.esat.kuleuven.be/ecrypt/courses/end/slides-28/6-kalker.pdf, May 2008. (pp. 18 and 19.)

[226] OpenID. OpenID official site. http://openid.net/. (p. 137.)

[227] Rafail Ostrovsky and William E. Skeith III. Private searching on streaming data. In *Advances in Cryptology - CRYPTO'05*, LNCS, pages 223–240. Springer-Verlag, 2005. (p. 77.)

[228] OWASP. Risk rating methodology. http://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology. (p. 70.)

[229] P3P. Platform for privacy preferences project: W3C P3P specifications. http://www.w3.org/TR/P3P/. (pp. 50 and 77.)

[230] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology - EUROCRYPT'99*, LNCS 1592, pages 223–238. Springer-Verlag, 1999. (pp. 77, 101, 102, 109, 119, 127, and 209.)

[231] Sameer Patil and Alfred Kobsa. *Privacy Considerations in Awareness Systems: Designing with Privacy in Mind*, chapter 8, pages 187–206. Human-Computer Interaction Series. Springer London, June 2009. (pp. 49, 78, and 83.)

[232] Shelby Pereira, Sviatoslav Voloshynovskiy, Maribel Madueno, Stéphane Marchand-Maillet, and Thierry Pun. Second generation benchmarking and application oriented evaluation. In *International Workshop on Information Hiding*, LNCS, pages 340–353. Springer-Verlag, 2001. (p. 170.)

[233] Juan Carlos Perez. Facebook's Beacon more intrusive than previously thought. http://www.pcworld.com/article/140182/facebooks_beacon_more_intrusive_than_previously_thought.html, 2007. (p. 3.)

[234] Fabien A. P. Petitcolas, Martin Steinebachb, Frédéric Raynal, Jana Dittmann, Caroline Fontained, and Nazim Fatès. A public automated web-based evaluation service for watermarking schemes: Stirmark benchmark. In *SPIE International Symposium on Electronic Imaging 2001*, volume 4314 of *Proceedings of the SPIE*, pages 575–584, 2001. (p. 170.)

[235] Milan Petković, Claudine Conrado, Geert-Jan Schrijen, and Willem Jonker. Enhancing privacy for digital rights management. In *Security, Privacy, and*

Trust in Modern Data Management, chapter 23, pages 347–364. Springer Berlin Heidelberg, June 2007. (pp. xxvii, 22, 23, and 24.)

[236] Liam Peyton, Jun Hu, Chintan Doshi, and Pierre Seguin. Addressing privacy in a federated identity management network for ehealth. In *Eighth World Congress on the Management of eBusiness WCMeB 2007*. IEEE Computer Society, July 2007. (p. 137.)

[237] Andreas Pfitzmann and Marit Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management (Version 0.33 April 2010). Technical report, TU Dresden and ULD Kiel, April 2010. `http://dud.inf.tu-dresden.de/Anon_Terminology.shtml`. (pp. 10, 11, 26, 44, 46, 47, 48, 52, and 176.)

[238] Andreas Pfitzmann, Birgit Pfitzmann, and Michael Waidner. ISDN-mixes: Untraceable communication with very small bandwidth overhead. In *Proceedings of the GI/ITG Conference on Communication in Distributed Systems*, pages 451–463, February 1991. (p. 76.)

[239] Andreas Pfitzmann and Michael Waidner. Networks without user observability – design options. In *Advances in Cryptology - EUROCRYPT'85*, LNCS 219, pages 245–253. Springer-Verlag, 1985. (p. 76.)

[240] Birgit Pfitzmann and Ahmad-Reza Sadeghi. Anonymous fingerprinting with direct non-repudiation. In *Advances in Cryptology - ASIACRYPT 2000*, LNCS 3376, pages 401–414. Springer-Verlag, 2000. (pp. 87 and 89.)

[241] Birgit Pfitzmann and Matthias Schunter. Asymmetric fingerprinting (extended abstract). In *Advances in Cryptology - EUROCRYPT'96*, LNCS, pages 84–95. Springer-Verlag, 1996. (pp. 87 and 88.)

[242] Birgit Pfitzmann and Michael Waidner. Anonymous fingerprinting. In *Advances in Cryptology - EUROCRYPT'97*, LNCS, pages 88–102. Springer-Verlag, 1997. (pp. 88 and 89.)

[243] Benny Pinkas. Cryptographic techniques for privacy preserving data mining. *SIGKDD Explorations*, 4(2):12–19, 2002. (p. 77.)

[244] Klaus Pommerening and Mainz. Michael Reng. Secondary use of the ehr via pseudonymisation. In L. Bos, S. Laxminarayan, and A. Marsh, editors, *Medical Care Compunetics 1: Studies in Health Technology and Informatics*, volume 103, pages 441–446. IOS Press, Amsterdam, 2004. (p. 137.)

[245] Portlet. JSR-286 the Java Portlet API 2.0. `http://www.jcp.org/en/jsr/detail?id=268`, 2008. (p. 139.)

[246] Richard A. Posner. *The Economics Of Justice*. Harvard University Press, January 1981. (p. 4.)

[247] Huffington Post. Facebook's Zuckerberg says privacy no longer a "social norm". `http://www.huffingtonpost.com/2010/01/11/facebooks-zuckerberg-the_n_417969.html`, Jan. 2010. (p. 3.)

[248] Guillaume Poupard and Jacques Stern. Fair encryption of *RSA* keys. In *Advances in Cryptology - EUROCRYPT 2000*, LNCS 1807, pages 173–190. Springer-Verlag, 2000. (pp. 103, 119, and 127.)

[249] PrimeLife. PrimeLife – Privacy and Identity Management for Europe in Life. http://www.primelife.eu/. (pp. 79 and 138.)

[250] J. P. Prins, Z. Erkin, and R. L. Lagendijk. Anonymous fingerprinting with robust QIM watermarking techniques. *EURASIP Journal on Information Security*, 2007, Article ID 31340, 13 pages, 2007. (p. 123.)

[251] William L. Prosser. Privacy. *Califorina Law Review*, 383, 1960. (p. 9.)

[252] U Prove Technology. Microsoft U-Prove CTP. https://connect.microsoft.com/content/content.aspx?ContentID=12505&SiteID=642. (p. 137.)

[253] Lintian Qian and Klara Nahrstedt. Watermarking schemes and protocols for protecting rightful ownership and customer's rights. *Journal of Visual Communication and Image Representation*, 9(3):194–210, Sept. 1998. (p. 88.)

[254] Michael O. Rabin. How to exchange secrets by oblivious transfer. Technical report TR-81, Aiken Computation Laboratory, Harvard University, 1981. (p. 77.)

[255] Emily Raymond. HP developing picture jamming technology to block unwanted photographs. http://www.digitalcamerainfo.com/d/News.htmhttp://www.digitalcamerainfo.com/content/HP-Developing-Picture-Jamming-Technology-to-Block-Unwanted-Photographs-.htm, January 2005. (p. 161.)

[256] Michael Reiter and Aviel Rubin. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, June 1998. (p. 76.)

[257] Harold C. Relyea. *9/11 Commission Recommendations: A Civil Liberties Oversight Board*. CRS Report for Congress, August 2004. http://www.fas.org/irp/crs/RS21906.pdf. (p. 3.)

[258] Alfredo Rial, Mina Deng, Tiziano Bianchi, Alessandro Piva, and Bart Preneel. Anonymous buyer-seller watermarking protocols – formal definitions and security analysis. *IEEE Transactions on Information Forensics and Security, To appear*, page 11, 2010. (pp. 28, 77, and 95.)

[259] Michael Ripley, C. Brendan S. Traw, Steve Balogh, and Michael Reed. Content protection in the digital home. *Intel Technology Journal*, 06(4):49–56, November 2002. http://www.intel.com/technology/itj/2002/volume06issue04/art05_protection/vol6iss4_art05.pdf. (p. 21.)

[260] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems (reprint). *Commun. ACM*, 26(1):96–99, 1983. (p. 101.)

[261] Michael Roe. *Cryptography and Evidence*. PhD thesis, University of Cambridge, Clare College, 1997. (p. 47.)

[262] Bill Rosenblatt. Steganography revisited: watermarking comes in from the cold. *Seybold report: analyzing publishing technologies*, 3(5):3–6, June 2003. (pp. 170 and 172.)

[263] Tom Scavo and Scott Cantor. Shibboleth architecture, technical overview working draft 02. http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-tech-overview-latest.pdf, June 2005. (p. 137.)

[264] Bruce Schneier. *Secrets & Lies: Digital Security in a Networked World*. Wiley, 2000. (p. 41.)

[265] Bruce Schneier. Protecting privacy and liberty, nature 413, 773. `http://www.nature.com/nature/journal/v413/n6858/full/413773a0.html`, October 2001. (p. 5.)

[266] Bruce Schneier. Security vs. privacy. `http://www.schneier.com/blog/archives/2008/01/security_vs_pri.html`, January 2008. (p. 6.)

[267] Sensaura. UK companies team to solve worldwide camera phone privacy abuse. british safe haven technology enables digital cameras to be disabled in a localized environment. `http://www.cellular.co.za/news_2004/june/062204-uk_companies_team_to_solve_world.htm`. (p. 161.)

[268] Min-Hua Shao. A privacy-preserving buyer-seller watermarking protocol with semi-trust third party. In *Trust, Privacy and Security in Digital Business*, LNCS 4657, pages 44–53. Springer-Verlag, Aug. 2007. (pp. 90 and 91.)

[269] Koen Simoens, Pim Tuyls, and Bart Preneel. Privacy weaknesses in biometric sketches. In *Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*, pages 188–203. IEEE Computer Society, 2009. (p. 77.)

[270] Champskud J. Skrepth and Andreas Uhl. Robust hash functions for visual data: An experimental comparison. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 986–993, 2003. (p. 171.)

[271] Daniel J. Solove. Conceptualizing privacy. *California Law Review*, 90, 2002. (p. 8.)

[272] Daniel J. Solove. A taxonomy of privacy. *University of Pennsylvania Law Review*, 154(3), January 2006. `http://ssrn.com/abstract=667622`. (pp. 9, 26, 43, and 44.)

[273] Daniel J. Solove. I've got nothing to hide' and other misunderstandings of privacy. *San Diego Law Review*, 44, 2007. (pp. 4, 5, and 6.)

[274] Daniel J. Solove. *Understanding Privacy*. Harvard University Press, May 2008. (pp. 4, 8, 9, and 43.)

[275] Latanya Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):571–588, 2002. (pp. 77 and 80.)

[276] Latanya Sweeney. K-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, 2002. (pp. 60 and 77.)

[277] Dean Takahashi. Web users will trade privacy for security and convenience. `http://social.venturebeat.com/2009/09/15/web-users-will-trade-off-privacy-for-security-and-convenience/`, September 2009. (p. 5.)

[278] Gábor Tardos. Optimal probabilistic fingerprint codes. *J. ACM*, 55(2):116–125, 2008. (p. 104.)

[279] TAS3. Trusted Architecture for Securely Shared Services – TAS3. `http://www.tas3.eu/`. (pp. 135 and 138.)

[280] TCG. Trusted Computing Group. `http://www.trustedcomputinggroup.org/`. (pp. 169 and 185.)

[281] Wade Trappe, Min Wu, Z. Jane Wang, and K. J. Ray Liu. Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation. *IEEE Trans. on Image Processing*, 51(4):1069–1087, 2003. (p. 87.)

[282] Mejdi Trimeche and Fehmi Chebil. Digital rights management for visual content in mobile applications. In *First International Symposium on Control, Communications and Signal Processing*, pages 95–98. IEEE, March 2004. (p. 172.)

[283] Axel van Lamsweerde. *Requirements Engineering: From System Goals to UML Models to Software Specifications*. Wiley, 2009. (p. 33.)

[284] Axel van Lamsweerde, Simon Brohez, Renaud De Landtsheer, and David Janssens. From system goals to intruder anti-goals: Attack generation and resolution for security requirements engineering. In *Proceedings of the RE'03 Workshop on Requirements for High Assurance Systems (RHAS'03)*, pages 49–56, 2003. (p. 41.)

[285] Vassilios S. Verykios, Elisa Bertino, Igor Nai Fovino, Loredana Parasiliti Provenza, Yucel Saygin, and Yannis Theodoridis. State-of-the-art in privacy preserving data mining. *ACM SIGMOD Record*, 3(1):50–57, Mar. 2004. (p. 77.)

[286] Michael Waidner and Birgit Pfitzmann. The dining cryptographers in the disco: Unconditional sender and recipient untraceability. In *Advances in Cryptology - EUROCRYPT'91*, LNCS 434, page 23. Springer-Verlag, 1990. (p. 76.)

[287] Z. Jane Wang, Min Wu, H. Vicky Zhao, Wade Trappe, and K. J. Ray Liu. Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation. *IEEE Trans. on Image Processing*, 14(6):804–821, 2005. (p. 87.)

[288] Samuel Warren and Louis Brandeis. The right to privacy. *Harvard Law Review*, 4:193–220, 1890. (pp. 7 and 9.)

[289] Alan F. Westin. *Privacy and Freedom*. The Bodley Head Ltd, Atheneum, New York, April 1967. (pp. 4, 6, and 7.)

[290] Marko Wolf, Andre Osterhues, and Christian Stueble. Secure offline superdistribution for mobile platforms. *International Journal of Applied Cryptography*, 1(4):251–263, August 2009. (p. 17.)

[291] Raymond B. Wolfgang, Christine I. Podilchuk, and Edward J. Delp. Perceptual watermarks for digital images and video. *Proceedings of the IEEE: special issues on identification and protection of multimedia information*, 87(7):1108–1126, July 1999. (pp. 170 and 173.)

[292] Min Wu, Wade Trappe, Z. Jane Wang, and K. J. Ray Liu. Collusion resistant fingerprinting for multimedia. *IEEE Signal Processing Magazine*, 21(2):15–27, March 2004. (p. 173.)

[293] Kim Wuyts, Riccardo Scandariato, Geert Claeys, and Wouter Joosen. Hardening XDS-based architectures. In *Proceedings of the 2008 Third International Conference on Availability, Reliability and Security*, pages 18–25, March 2008. (p. 140.)

[294] Kim Wuyts, Riccardo Scandariato, Bart De Decker, and Wouter Joosen. Linking privacy solutions to developer goals. In *Proceeding of the International Conference on Availability, Reliability and Security*, pages 847–852. IEEE Computer Society, 2009. (pp. 36 and 75.)

[295] Andrew Chi-Chih Yao. Protocols for secure computations. In *Proceedings of Twenty-third IEEE Symposium on Foundations of Computer Science*, pages 160–164, November 1982. (p. 76.)

[296] Adam Young and Moti Yung. Auto-recoverable auto-certifiable cryptosystems. In *Advances in Cryptology - EUROCRYPT'98*, LNCS, pages 17–31. Springer-Verlag, 1998. (p. 103.)

[297] YouTube. Google CEO Eric Schmidt on privacy, retrieved 2010-04-30. `http://www.youtube.com/watch?v=A6e7wfDHzew`, December 2009. (p. 3.)

[298] Eric Yu and Luiz Marcio Cysneiros. Designing for privacy and other competing requirements. In *Proceedings of the 2nd Symposium on Requirements Engineering for Information Security*, pages 15–16, 2002. (pp. 34 and 183.)

[299] J. Zhang, W. Kou, and K. Fan. Secure buyer-seller watermarking protocol. *IEE Proceedings Information Security*, 153(1):15–18, Mar. 2006. (pp. 90 and 91.)

Curriculum Vitae

Mina Deng was born on September 19, 1981 in Beijing, China. She is currently a research assistant at the Computer Security and Industrial Cryptography (COSIC) research lab, department of Electrical Engineering, K.U.Leuven, Belgium. She received the MSc. degree, magna cum laude, in Electrical Engineering from the K.U.Leuven in 2004. She also works as a scientific researcher for the Interdisciplinary Institute for BroadBand Technology (IBBT) in Belgium.

She has been involved in a number of research projects, including "EU SPEED – Signal Processing in the Encrypted Domain" (2007-2010), "EU FIDIS – Future of Identity in the Information Society" (2004-2008), "EU ECRYPT – European Network of Excellence for Cryptology" (2004-2008), "EU RE-TRUST – Remote EnTrusting by RUn-time Software authentication" (2008-2009), "IBBT IPEA – Innovative Platform for Electronic Archiving" (2005-2007), "IBBT E-HIP – E-Health Information Platforms" (2006-2008), "IBBT Share4Health – Healthcare professional's collaboration Space" (2008-2010), "IBBT AQUA – Automated Quality assessment and Authentication based on watermarking and perceptual hashing" (2010-present), and "GOA AMBioRICS – Algorithms for Medical and Biological Research, Integration, Computation and Software" (2007-2010).

She received "Outstanding Achievement Award for best student research paper", granted by the New Zealand State Services Commission in 2008, for the joint work with Danny De Cock entitled "Towards cross-context identity management framework in e-health"; "Best Student Paper Award" of the IEEE International Symposium on Electronic Commerce and Security 2008, for the paper "Attacks on two buyer-seller watermarking protocols and an improvement for revocable anonymity"; and the "Barco/VIK-Prize" as one of the best engineering theses in Belgium in 2003.

List of Publications

International Journals

1. Mina Deng, Danny De Cock, and Bart Preneel. Towards a cross-context identity management framework in e-health. *Online Information Review*, 33(3):422-442, 2009.

2. Mina Deng and Bart Preneel. Attacks on two buyer-seller watermarking protocols and an improvement for revocable anonymity. *International Journal of Intelligent Information Technology Application*, 1(2):53-64, 2008.

3. Mina Deng, Kim Wuyts, Riccardo Scandariato, Bart Preneel, and Wouter Joosen. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirement Engineering Journal special issue on Data Privacy*, to appear, 27 pages, 2010.

4. Alfredo Rial, Mina Deng, Tiziano Bianchi, Alessandro Piva, and Bart Preneel. Anonymous buyer-seller watermarking protocols – formal definitions and security analysis. *IEEE Transactions on Information Forensics and Security*, to appear, 11 pages, 2010.

Book Chapters

1. Mina Deng, Danny De Cock, and Bart Preneel. An interoperable cross-context architecture to manage distributed personal e-health information. In M. M. Cunha, R. Simoes, and A. Tavares, editors, *Handbook of Research on Developments in e- Health and Telemedicine: Technological and Social Perspectives*, ISBN: 978-1-61520-670-4, chapter 27, pages 576-602. Hershey, PA, USA: IGI Global, Inc., 2009.

2. Mina Deng and Bart Preneel. On secure buyer-seller watermarking protocols with revocable anonymity. In Kyeong Kang, editor, *E-Commerce*, ISBN: 978-953-7619-98-5, chapter 11, pages 184-202. IN-TECH Education and Publishing, Vienna, Austria, 2009.

International Conferences

1. Mina Deng, Lothar Fritsch, and Klaus Kursawe. Personal rights management – enabling privacy rights in digital online content. In Jana Dittmann, Stefan Katzenbeisser, and Andreas Uhl, editors, *Communications and Multimedia Security, 9th IFIP TC-6 TC-11 International Conference (CMS)*, LNCS 3677, pages 266-268, Salzburg, Austria. Springer, 2005.

2. Brecht Wyseur, Karel Wouters, Mina Deng, Thomas Herlea, and Bart Preneel. On the Design of a Secure Multimedia Archive. In *1st Benelux Workshop on Information and System Security (WISSec 2006)*, pages 1-14, Antwerp, Belgium, 2009.

3. Mina Deng, Lothar Fritsch, and Klaus Kursawe. Personal rights management – taming camera-phones for individual privacy enforcement. In George Danezis and Philippe Golle, editors, *Privacy Enhancing Technologies, 6th International Workshop (PET), Revised Selected Papers*, LNCS 4258, pages 172-189, Cambridge, UK. Springer, 2006.

4. Mina Deng and Danny De Cock. Towards cross-context identity management framework in e-health. In *Managing Identity in New Zealand - Identity Conference*, 10 pages, Wellington, New Zealand, 2008.

5. Mina Deng and Bart Preneel. On secure and anonymous buyer-seller watermarking protocol. In Abdelhamid Mellouk, Jun Bi, Guadalupe Ortiz, Dickson K. W. Chiu, and Manuela Popescu, editors, *Third International Conference on Internet and Web Applications and Services (ICIW)*, pages 524-529, Athens, Greece. IEEE Computer Society, 2008.

6. Mina Deng and Bart Preneel. Attacks on two buyer-seller watermarking protocols and an improvement for revocable anonymity. In Fei Yu, Qi Luo, Yongjun Chen, and Zhigang Chen, editors, *Proceedings of The International Symposium on Electronic Commerce and Security (ISECS)*, pages 923-929, Guangzhou, China. IEEE Computer Society, 2008.

7. Mina Deng, Li Weng, and Bart Preneel. Anonymous buyer-seller watermarking protocol wth additive homomorphism. In Pedro A. Amado Assunção and Sérgio M. M. de Faria, editors, *Proceedings of the International Conference on Signal Processing and Multimedia Applications (SIGMAP)*, pages 300-307, Porto, Portugal. INSTICC Press, 2008.

8. Mina Deng, Riccardo Scandariato, Danny De Cock, Bart Preneel, and Wouter Joosen. Identity in federated electronic healthcare. *In 1st IFIP Wireless Days (WD) - the 6th IFIP Network Control conference (Netcon)*, pages 1-5, Dubai, UAE. IEEE Computer Society, 2008.

9. Mina Deng, Tiziano Bianchi, Alessandro Piva, and Bart Preneel. An efficient buyer-seller watermarking protocol based on composite signal representation. In *Proceedings of the 11th ACM workshop on Multimedia and security (MMSEC)*, pages 9-18, Princeton, New Jersey, USA. ACM New York, NY, USA, 2009.

10. Mina Deng, Tiziano Bianchi, Alessandro Piva, and Bart Preneel. Efficient implementation of a buyer-seller watermarking protocol using a composite signal representation. In J. Guajardo and A. Piva, editors, *The International Workshop on Signal Processing in the EncryptEd Domain (SPEED 2009)*, pages 22-41, Lausanne, Switzerland, 2009.

Project Deliverable

1. Mina Deng. Multimedia encryption in electronic archiving. *Deliverable 6.1: IPEA – Security components in electronic archiving*, 47 pages, 2006.

2. Mina Deng. Digital watermarking in electronic archiving. *Deliverable 6.2: IPEA – Security components in electronic archiving*, 32 pages, 2006.

3. Mina Deng, Claudia Diaz, Els Soenens. Weblogs, Privacy and Profiling. *contribution to FIDIS WP7 Profiling*, 9 pages, March 2006.

4. Mina Deng, Claudia Diaz, Nessim Kisserli, Stefan Schiffner, Carmela Troncoso, Karel Wouters, and Antoon Bosselaers. A survey of privacy enhancing techniques for e-Health, *Deliverable 3.4.1: E-HIP – E-Health Information Platforms*, 63 pages, 2007.

5. Mina Deng, and Danny De Cock. Proposed privacy solution for e-Health information platforms. *Deliverable 3.4.2: E-HIP – E-Health Information Platforms*, 39 pages, 2007.

6. Danny De Cock, Mina Deng, Sebastian Faust, Svetla Nikova, Dries Schellekens, Dennis Vandevenne, and Karel Wouters. Applicability of e-ID and alternative identification mechanisms. *Deliverable 3.2.1: E-HIP – E-Health Information Platforms*, 48 pages, 2007.

7. Mina Deng, and Danny De Cock. Feasibility of federated identity and roadmap for integration in e-Health. *Deliverable 3.2.2: E-HIP – E-Health Information Platforms*, 32 pages, 2007.

8. Brecht Wyseur, Mina Deng, and Thomas Herlea. A Survey of Homomorphic Encryption Schemes. *Deliverable of RE-TRUST – Remote EnTrusting by RUn-time Software authentication.* 15 pages, 2007.

9. Mina Deng. Privacy and data protection requirements for Share4Health. *Deliverable 1.2.1: Share4Health – Healthcare professional's collaboration Space*, 21 pages, 2009.

10. Mina Deng. SPEED scenario on buyer-seller watermarking protocols. *Deliverable 4.2: SPEED – Signal Processing in the Encrypted Domain*, 12 pages, 2009.

11. Mina Deng, and Bart Preneel. Privacy and data protection architecture, *Deliverable 1.2.2: Share4Health – Healthcare professional's collaboration Space*, 36 pages, 2009.

12. Mina Deng, and Bart Preneel. Digital identities lifecycle management, *Deliverable 1.4.1: Share4Health – Healthcare professional's collaboration Space*, 19 pages, 2010.

13. Mina Deng, and Dieter Bardyn. State-of-the-art, requirements and high-level architecture for watermarking and perceptual hashing for quality assessment and authentication, *Deliverable 2.1: AQUA – Automated Quality assessment and Authentication based on watermarking and perceptual hashing*, 32 pages, 2010.

14. Mina Deng, Tiziano Bianchi, Alessandro Piva, Alfredo Rial, and Bart Preneel. Anonymous buyer-seller watermarking protocols – efficient implementations, Technical Report, 11 pages, 2010.

Arenberg Doctoral School of Science, Engineering & Technology
Faculty of Engineering
Department of Electrical Engineering (ESAT)
Research group ESAT/SCD
Kasteelpark Arenberg 10 Box 2446
B-3001 Leuven-Heverlee Belgium